INTERACTIVE SIMULATION OF AIR COMBAT

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of

MASTER OF TECHNOLOGY

by

SQN. LDR. J. N. RAMPAL

to the

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY, KANPUR

FEBRUARY, 1987

CERTIFICATE

Certified that this work on 'Interactive Simulation of Air Combat' by Squadron Leader J.N. Rampal has been carried out under our supervision and that this has not been submitted elsewhere for a degree.

Vishwanath Sinha

(Vishwanath Sinha)
Professor
Electrical Engineering Deptt.
Indian Institute of Technology
Kanpur

Dhande

(S.G. Dhande)
Professor
Dept. of Computer Science and Engg.
Indian Institute of Techno
Kanpur

Feb 9, 1987

ACKNOWLEDGEMENTS

I wish to convey my sincere gratitude to Prof. S.G. Dhande and Prof. V. Sinha for their advice, guidance and encouragement during my project work. Special thanks are due to Prof. Sinha for his help with Fuzzy Set theoretic formulations and modelling; and to Prof. Dhande for introducing me to the fascinating world of Interactive computer graphics.

I owe thanks to my fellow-students, whose association is responsible for making my 'M.Tech.' studies a memorable experience. A 'thank you' to the exceptional team of research engineers at CAD-P centre and ever-helpful CAD-P staff for lending support whenever required. Thanks to Mr. J.S. Rawat for transforming handwritten manuscript to the printed form.

Finally my thankful appreciation of my wife's remarkable display of patience with me and my late hours.

Sqn.Ldr. J.N. Rampal

ABSTRACT

Air combat simulators are used by advanced airforces of the world for tactics evaluation and/or training. The systems are generally complex and use dedicated hardware/ software to implement the graphic view generation, interaction handling and computation/supervisory functions.

The present work is intended to be a simplistic version of the above. All the three functions are software implemented on the same general purpose computer. Consequently the simulation of various functions is simplified and speed of simulation is also limited.

The modelling of the aircraft flight follows from the general flight principles. The model is made simple by putting limits on manoeuvres and assuming proportional relationships. The war-scenario interaction models follow from the general air war-gaming principles. The weapon interaction is solved by associating graded-damage membership functions of the fuzzy set theoretic modelling with damage assessment calculations. The view generation is by creating overlapping segments by using PLOT-10 GKS routines and then making the segments selectively visible to get the required

view. The interaction between the two participants is handled through a common file, in which, each of the interactive terminal can read or write. The local interaction is monitored by periodically scanning the keyboard buffer and reading input, if any.

The simulation uses discrete event activity scanning approach. Emphasis in the work is on synthetic view generation. Effectiveness of the weapon interaction, implementation has been checked by a simulation run on synthetic data. The project work makes a simplified air-combat simulator for 'one versus one' scenario.

TABLE OF CONTENTS

Page

Page

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

This work is a non-probabilistic modelling and interactive, two terminal simulation of 'one versus one' aircombat. The endeavour is to present, graphically, a pilot's view of a 'dogfight'. The combat has been mathematically modelled using wargaming techniques. Fuzzy set theoretic approach has been applied to solve weapon interaction. Software techniques have been used to simulate pilot's actions as well as generation of cockpit and outside world views. The program implementation is on a NORSK-DATA/560CX computer in NORSK-DATA superset of FORTRAN-77. The graphic view generation uses PLOT-10 GKS library routines, on two TEKTRONIX-4109 graphic terminals.

Inspiration for the work has been drawn from the 'AIR COMBAT SIMULATORS' in operation with some advanced Airforces of the world. One such simulator is known to be operating with ROYAL AIR FORCE. But apart from their existence, announced from time to time in military and aviation journals like INTERAVIA and INTERNATIONAL DEFENCE REVIEW, not much is known about these systems due to classified nature of the projects. Fast obsolescence of military fighter aircraft

and long gestation periods for development of such systems for an aircraft type, put enormous constraints on the research and development in this field.

1.1 AIRCRAFT SIMULATOR TYPES AND DISTINCTIONS BETWEEN THE TYPES

An aircraft simulator can be one of the two types

i) Landing aircraft simulator

ii) Air combat simulator

The two types differ in the segment of the flight profile, which is simulated. The landing aircraft simulator simulates the flight of a single landing aircraft within the space in vicinity of a landing strip. The simulator operates 'on circuit' i.e. from a low level flight point to touch down on a landing strip.

An 'aircombat simulator' is basically different from the former type. It operates in a much larger 'combat space' and generally involves more than one aircraft. (Some EW simulators may operate with one aircraft). Interaction between the two aircraft is an important part of the simulation and it's specification must relate satisfactorily to the real life interaction. Also, to be an alternative to experience of real-flight, the operating stations should be as

representative as possible, of the actual cockpit layout, in terms of view presented and handling of controls despite being synthesised by the simulation.

The development of such a system, having replicate cockpit controls, will, in general, be a large project, in terms of cost, hardware and effort. The present work endeavours to provide only the view. Cockpit control inputs are simulated by keyboard operations. Also to keep the simulation from being complicated, the synthetic view does not present all the information, which will generally be available in the cockpit of any aircraft type. But it includes sufficient information to enable the simulation user to fly his 'sorties'.

1.2 USERS' VIEW OF SIMULATION

This simulation is intended to be used simultaneously by two users for the purpose of training or strategy design by either. One user is on an offensive mission and the other on an 'interceptor sortie'. Their respective objectives in terms of missions can be specified as follows.

1.2.1 Attacker's Mission:

i) To take off and enter the 400 km x 400 Km area belonging to the enemy (Automatic placement at the edge of enemy area provided by simulation).

ii) To manoeuvre and bomb enemy airfield situated in the middle of the defined enemy area (co-ordinates 200 Km, 200 km).

iii) To avoid being intercepted.

iv) To shoot down the interceptor if the combat ensues.

1.2.2 Defender's Mission:

i) To await attackers intrusion in the friendly area

ii) To keep the attacker away from the airfield

iii) To shoot down the attacker if air combat ensues.

Detailed specifications of the simulation are enumerated in system design discussion in Chapter IV. Also described therein, is the information that can be used by the simulation user to control the flight of his aircraft and to track the position of enemy aircraft.

1.3 APPROACH IN SYSTEM DESIGN AND SCOPE OF PRESENT WORK

No pre-requisites have been assumed for the design of this simulation. The problem formulation has been derived from the basic 'Airwar gaming' strategies. The flight envelopes of the two adversary aircraft, with all the manoeuvres, are computed using 'theory of flight' formulae in simplified forms. Fuzzy set theoretic approach has been

followed to compute damage assessment subsequent to weapon release. The overlapping graphic segments make up the synthetic cockpit view. To get a realistic view of continuous flight and combat, the view is modified or regenerated after each recomputation cycle.

The scope of this simulation is limited to presenting rudimentary level of 'air combat' process due to constraints on development time and hardware resources available. An aircraft handling simulation in real time, itself, could be requiring a mini-computer's computational ability. Similar hardware support is required for view generation and interaction/supervision functions. Real-world simulations use dedicated computers for each of these functions. For this work, therefore, it has required to present a simplified view of these functions, so as to present a real-time view of the combat.

Before describing the system design and implementation, it is in order to describe some of the underlying principles, techniques and theoretical background for this simulation. For the purpose, Chapter II describes War gaming and it's applications, past and present airwar gaming models, simulation techniques and interactive computer graphics

principles. Chapter III describes mathematical modelling of flight process and fuzzy set theoretic approach for solving weapon interaction. System design description follows in Chapter IV. Conclusions and suggestions for further work are enumerated in Chapter V.

To introduce the novice to military parlance, a glossary of terms with military connotations is placed as Appendix 'A'.

CHAPTER II

BASIC CONCEPTS

2.1 'WARGAMES' - AN OVERVIEW

A 'war-game' is presentation of the conduct of a war or it's component battles, by artificial means. The method of presentation may be mechanical, graphic or by a computer aided simulation. The purpose of constructing a war game could be one or more of the following:

i) To test a battle strategy

ii) To ascertain the effectiveness of a weapon system

iii) To train combatants

iv) for entertainment

A look at the history of 'war gaming' can give one an insight into the evolution of techniques/hardware and growth of applications of 'war-gaming' to military sciences. A brief chronological overview is, therefore, presented below.

'War-games' originated as a hobby for the War buffs. The earliest of the war games date back to the 'pre-World War-I' period. War sciences and weapon systems were not formally developed at that time; so the 'war games', too, were mechanical 'tinplate soldier and musket' models.

These, generally, depicted infantary battles. The 'game' would, generally, start with initial batallion/company deployment on a 'table top' terrain and the 'game players' manoeuvred their forces in accordance with laid down rules to represent real-life military actions. The weapons would be fired, casualities effected and game results manually computed using rules derived from existing military knowledge. During the WORLD WAR-II period, the 'war games' had announced their utility to military planners as a decision taking/ training tool; providing an impetus to work in this field.

The 'air war' games received attention only after WORLD WAR-I. During the WORLD WAR II the interdependence and effectiveness of each arm was an established fact. The 'war game' models, to reflect this fact, incorporated air and naval elements. The later models incorporated statistical methods to include chance elements for game result computations.

The arrival of digital computer in the post war period, also marked the coming of age of 'War gaming'. Use of computers has expanded the field of applicability of 'War games' to the whole gamut of military activity. The digital computer has not come to 'war gaming' too soon. The frantic pace of advancement of military technology and

specialisation of military functions/weapon systems/delivery vehicles have increased the number of variables in any military activity. The consequent complexity of military system, today, is not amenable to representation by mechanical models. In addition, the digital computers have made possible what could not be attempted earlier. A few examples are pre-prototype evaluation of systems, response of system under conditions which could not conveniently be created around actual systems. Operational,research on military planning and strategic trade-off evaluation could not be attempted without the computers; nor would be the valuable aid of 'system simulators' available without these. In fact the pace of development of new military weapons/strategies and their management could not be effected without digital computers.

Out of the above-mentioned applications, 'War games' include ones which generate battle scenario for the purpose of training or evaluation of weapon system/strategies. The evaluation of a strategy, in itself, may be a training process for the user; so it is difficult to make a distinction between the two types. Whatever be the application, the computer aided implementations of 'war-games' use similar techniques and practices.

The mechanical models, presently, have limited use and are mainly a hobby item. The computer based 'war gaming', in contrast, has developed into a complex activity requiring large resources. Governments and large companies support 'war-gaming' projects for military application/promoting and assessing the potential of their products respectively. Banerjee [1] has compiled a list of such projects. The projects and the organisations under-taking them are enumerated on the next page. The list is only representative, In fact, in all western countries, all major weapon system/aircraft manufacturers and defence researchers have 'war-gaming' or simulation studies for evaluation, proving and development of their products.

2.2 AIRWAR GAME MODELS: PAST AND PRESENT

Present day airwar operations include the following activities

i) Strategic Reconnaissance

ii) Strategic Bombing

iii) Air Superiority

iv) Close Air Support

v) Interception

vi) Inter-diction

vii) Transport Support and Casuality Evacuation

Table 2.1: List of some War Games Projects

S.No.	Organization	Brief project description
1.	Defence Research and Analysis Establishment (DRAE), Department of National Defence (Canada)	1. Strategic Air Defence 2. Ground air Co-operation 3. Weapon system allocation 4. Training war games 5. Air Defence of the sea-line
2.	Operation Research Gruppen (FRG)	1. Dynamic Combat Analysis 2. Command, communication and Information System 3. Scenario Studies 4. Support systems
3.	SROAT: Army research Group (FRANCE)	1. Weapon System Definition and usage
4.	SROM: Navy research Group (FRANCE)	1. Weapon evaluation 2. Tactics Evaluation 3. Statistical Evaluation (Tarpedo launch analysis)
5.	Defence Operation Analysis Organization (DOAO) (UK)	1. NATO air defence 2. Close air support 3. Anti submarine warfare 4. Tank-exposure time 5. Weapon systems
6.	Strategic and Tactical Analysis Group, (USA)	1. 'LEGION' - Land warfare war game 2. 'CASCADE' - Computerised air-strike and air defence evaluation 3. 'ORION' - Logistic network and interdiction system

viii) Tactical Surveillance and Fire Control

ix) Electronic Surveillance

x) Electronic Counter Measures

Earlier models represented air war operations of corresponding periods, so were concerned with activities listed at serial (iv), (v) and (vi). The other activities, have followed from development of higher performance aircraft and arrival of Electronic war-fare.

Describing a few models of each type, would be instructive to know the hardware techniques and practices in implementing the 'war-games'. But to keep the discussion short, the important features of each type are mentioned.

2.2.1 Earlier Models:

The earlier 'airwar games' were table-top models, where the horizontal top of the table or room floor served as simulated terrain and features/installations were drawn/ placed on it, using a scaled down grid coordinate system to define positions. Scaled down models of airplanes, hung by wires/supported on stands 'flew' above this scenario. The airplanes moved by wires/stands in horizontal and vertical directions and movement in this 'Combat space' was governed by specified rules so as to represent factual aircraft parameters associated with aircraft operational at the

corresponding times. The models were physically manipulated and interaction between weapon/target or weapon/aircraft manually computed according to pre-determined rules. The information exchange was controlled to represent real-life scenario e.g. an aircraft could see another, if the intervening distance were within some limits.

Evidently, these models, had viable applications only in WORLD WAR II situations. Also a number of factors, such as pilot's skills, accurate positioning and attitude determination for the aircraft could not be incorporated. Despite limitations, these 'war-games' did play a useful role till computer provided a break through in 'War-gaming' science.

A few such models are described by FEATHERSTONE [2]. 'Flecher-Pratt Principles' in defining modelling as described by FEATHERSTONE [2] could be adapted for mathematical modelling (with some modifications) for computer applications also.

2.2.2 Present day Models:

With the introduction of multi-role aircraft, high performance aircraft, radars, electronic warfare and numerous sophisticated command and control systems, 'airwar game' has become un-representable by mechanical models. Even

otherwise the 'war-gamer' for defence applications has felt the need of presenting a more detailed analysis of 'airwar activity' than is possible with mechanical models.

The computer based 'war-games' are the response to the above. The present day 'airwar games' are the simulations of airwar on digital computers. The outputs of the simulation are presented as real-life feed-back. Synthetic cockpits/ control panels using digital/analog display components are replicas of actual systems. The 'airwar' activities are mathematically modelled to include as many characteristics of the activity as desired. The quality of representation is limited by what one can afford in terms of costs and time. These improvements and increase in range of application of 'airwar gaming' can be assessed from the following few examples which enumerate the activities taken up for 'war-game' studies

i) Air Combat

ii) EW simulations for ECM and ECCM tactics evaluation

iii) Close Air Support Operations

iv) 'CASCADE' - Computerised Air-strike and counter air-defence evaluation

v) Air transport operations

The simulation techniques are extensively applied to represent 'airwar' activities and systems in 'airwar games'. The various techniques and their applicability to specific class of problems is described in the following section.

2.3 SIMULATION

EVANS [3] defines simulation as 'imitating', 'representing', 'appearing' or 'give the effect of', something else. Though simulation is an entity in itself, it's meaning is always related to some other activity or system. A simulation may employ abstract (mathematical), mechanical, electrical (analog) or digital computer methods for representation of the activity or system. A mathematical model takes in variations in the parameters and variables as input, and then mathematical manipulations can compute response according to pre-defined procedures or formulations. Irving Fischer's physical model of market economy, as described by BARTON [6] is a mechanical model making use of tubes, cisterns, stoppers and levers to simulate interaction of economic parameters. Electrical machines have their analog models using op-amps and other circuit elements. But the following discussion focusses on digital computer simulation only.

Firstly, what is the desirability of constructing a simulation? To study an existent/non-existent system, to

find out its viability/characteristics/limitations, an analytic approach of analysis is always preferable. But if the process/system has a very large number of variables/ inter-relationships or has stochastic variation built into the system, the analytic approach is severly limited. In such cases, digital computer simulation is the most appropriate and cost effective approach for analysis. Also, a system which is still in conceptual stage, can be modelled and run as a simulation. MARTIN [4] has described some advantages, salient ones are as follows:

i) Decisions concerning systems in conceptual phase can be made. Choices between systems can be exercised without actually employing resources on building systems.

ii) System performance can be simulated and observed under all conceivable conditions. System and environmental parameters changed to test systems to operable limits. With real-systems, it may not be be possible to operate them under 'impossible to achieve', conditions.

iii) System trials and evaluations can be speeded up/ slowed down by orders of magnitude. Thus a slow process can be quickly analysed and a very fast process slowed down for accurate observations.

iv) Only in a computer simulation, a snapshot of system state can be recorded at desired time. Also a simulation can be halted and restarted from the same point. Real system observations are 'on the fly'; and it will be generally impossible to halt/restart in the middle of operations.

v) Synthetic data can be generated for exercising real world systems.

vi) Simulation is a safe technique for testing hazardous systems. e.g. nuclear reactors

vii) Simulation is a much safer and cost effective training method rather than employing real-systems for the purpose. A landing aircraft simulator crashes may times, while training ab-initio pilots. One could not afford to use real aircraft for the purpose. Also the systems could be exercised to limits, otherwise, unacceptable due to safety considerations.

2.3.1 SIMULATION EXAMPLES

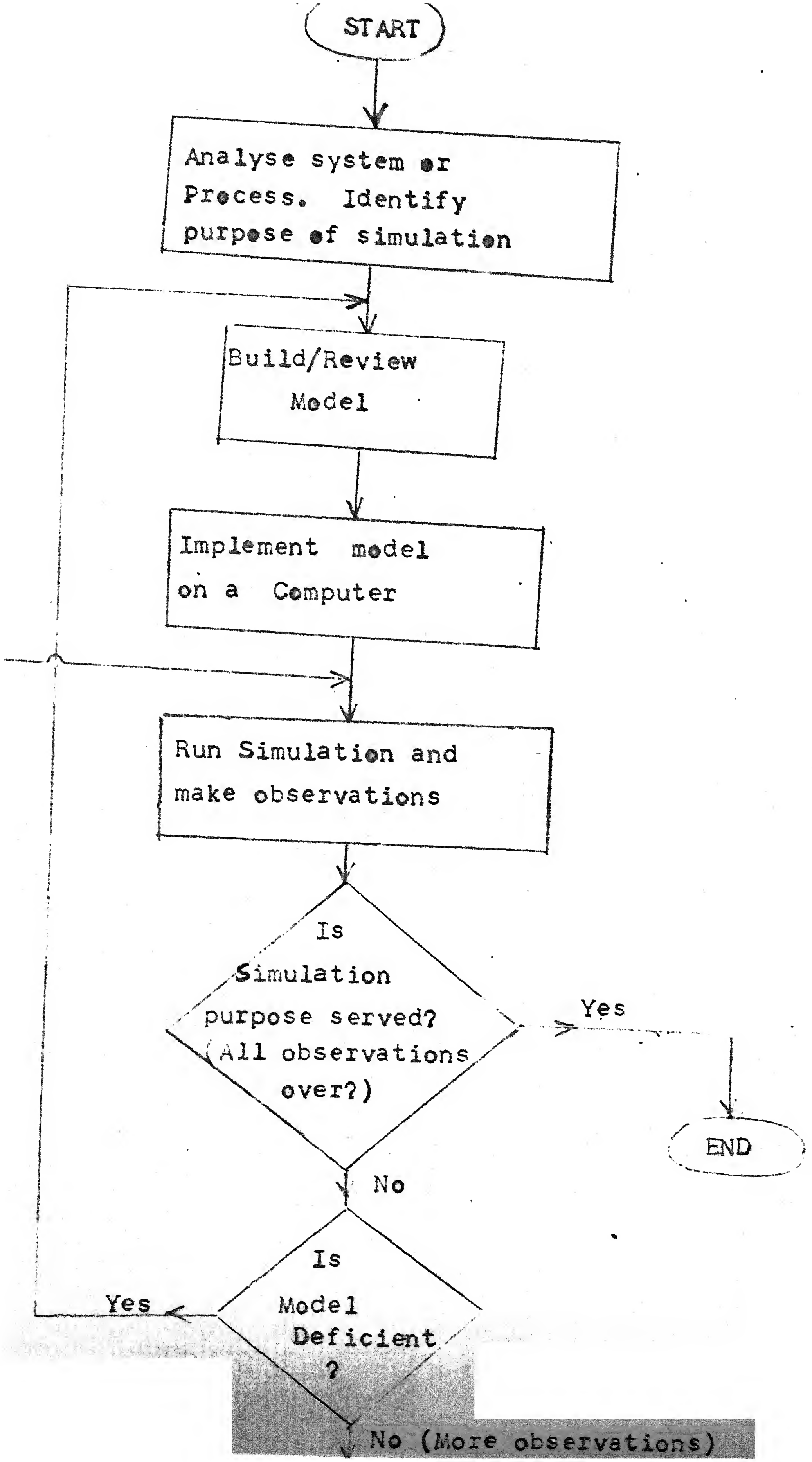
Use of simulation to avail above-mentioned advantages, is evident from the following examples of uses of simulation as described by MARTIN [4] and BARTON [6].

1. Simulations to optimise aircraft design
2. Simulation to evaluate effectiveness and response time of Air-defence networks
3. Simulations to design communication network topography for optimum efficiency.
4. Simulations to devise business strategies
5. Simulations to evaluate systems in their conceptual phase.
6. War-gaming to analyse military strategies and to train commanders in tactics.

The last mentioned application is a fast growing area. As the weapon systems become more complex and sophisticated; and hence more costly, their design, evaluation and further development becomes expensive in terms of cost and time. Simulation, then, becomes even more cost effective method of analysis and evaluation.

2.3.2 Simulation Methodology:

To construct a simulation, the system or process to be simulated is first analysed and purpose and scope of simulation is defined. One specifies the attributes of the system or process which are to be monitored. A model of the simulation is then built. It is translated into the computer program and simulation is then, ready to run.


START
Analyse system or Process. Identify purpose of simulation
Build/Review Model
Implement model on a Computer
Run Simulation and make observations
Is Simulation purpose served? (All observations over?)
Yes
END
No
Is Model Deficient ?
Yes
No (More observations)

A variety of techniques and tools exist, and are being developed for modelling and simulation programming. A system can be represented by many a models, differing in depth of detail, relative importance of component activities and their inter-relationships. Generally there is a trade-off between the cost of detail and information from simulation. An analysis/designer selects the viable synthesis of the two.

Special simulation languages and techniques have been developed. Specific application areas have b seen development of dedicated hardware also for the simulation purposes.

Development of a simulation can be represented by a flow chart as shown on the next page. No specifics are assumed to the nature of simulation or techniques; so is representative of any simulation's development.

2.3.3 Types of Simulation:

FISHMAN [5] has described various types of simulations. Depending upon quantum of abstract and real world components of systems, it can be classified under three types.

i) IDENTITY SIMULATION: System itself is used for running combinations of input parameters. This has no advantages of computer simulations and is very limited in system response analysis. Also it is seldom feasible and expensive.

ii) QUASI-IDENTITY SIMULATION: Here synthetic inputs could be exercising the automatic part of the system e.g. friendly aircraft could act as ghost intruders to assess detection and response characteristics of an air-defence system.

iii) LABORATORY SIMULATION: This is the most cost effective and feasible type of simulation. Additional advantages are flexibility of usage, portability of system, recording facilities, possibility of time scaling, replication, stopping and restarting at in-between time during the runs.

2.3.4 Simulation of Continuous and Discrete Systems:

Distinction need be made between the basic modelling techniques to simulate continuous and discrete systems.

A continuous system state changes continuously with time whereas a discrete system state changes at specified time points (called events in time). The discrete system state is assumed to be constant in the inter-event interval. SIMULATION REFERENCE [7] describes the methodology of simulating such systems.

A continuous system is modelled as a set of differential equations, which are evaluated after every Δt seconds (Δt being a very small interval). A chemical reaction,

nuclear fission, projectile path analysis are such applications.

Discrete systems are modelled as a set of inter-relationships which can be computed every time an event takes place. The computations ascribe a new set of values to the parameters which can be recomputed on next event. The appropriate technique for modelling a particular system depends upon the nature of the inter-event intervals. These intervals may be random or deterministic. When these are random, the modelling technique must allow for varying length of the intervals. When the intervals are deterministic, these may vary according to a plan or may be of equal length. The second may, generally, allow simplification of the model.

In discrete models, the inactivity during the inter-event interval need not be modelled so after all state changes have been made, one may advance to the time of next event for computing state changes corresponding to next event. This is termed 'next event approach to time advance'.

Another approach is to advance time by a fixed amount. This is called 'period modelling' and could be used for real-time simulation or 'scaled real-time' simulations.

2.4 INTERACTIVE SIMULATION

2.4.1 Need for Interactive Feature:

In semi-automatic systems, human beings form the control component of the system. They continuously take decision to control system operation. A manager in an industrial unit, a military commander in the battle field or an airtraffic controller on an airfield continuously view the situation, assess the required change in system parameters and give directions so as to put the system in a desired state of operation.

If such a system were to be modelled for simulation, the most complex component (of the system) to model would be the decision taking process of the human mind. A human mind's capabilities will, perhaps, be imitated to a limited extent in the coming years. Even so, the computational requirements for such a capability are immense. So the choices in simulation are either to invest large computational resources for modelling 'decision taking' or approximate it.

Another possibility is to 'Identity Simulate' the human being and computer simulate rest of the system. The system simulation operates till human interaction is warranted. Then the 'Identity Simulated' human can be prompted for

interaction. Interaction could be interrupt based or polled. Such a simulation has the additional advantage of training the operator in control procedures.

2.4.2 Design of an Interactive Simulation:

To incorporate the above-mentioned control configuration in a simulation, the hardware need be configured as a closed feed-back loop as shown in Fig. 2.2. Such a simulation will be called an INTERACTIVE SIMULATION. When the simulation

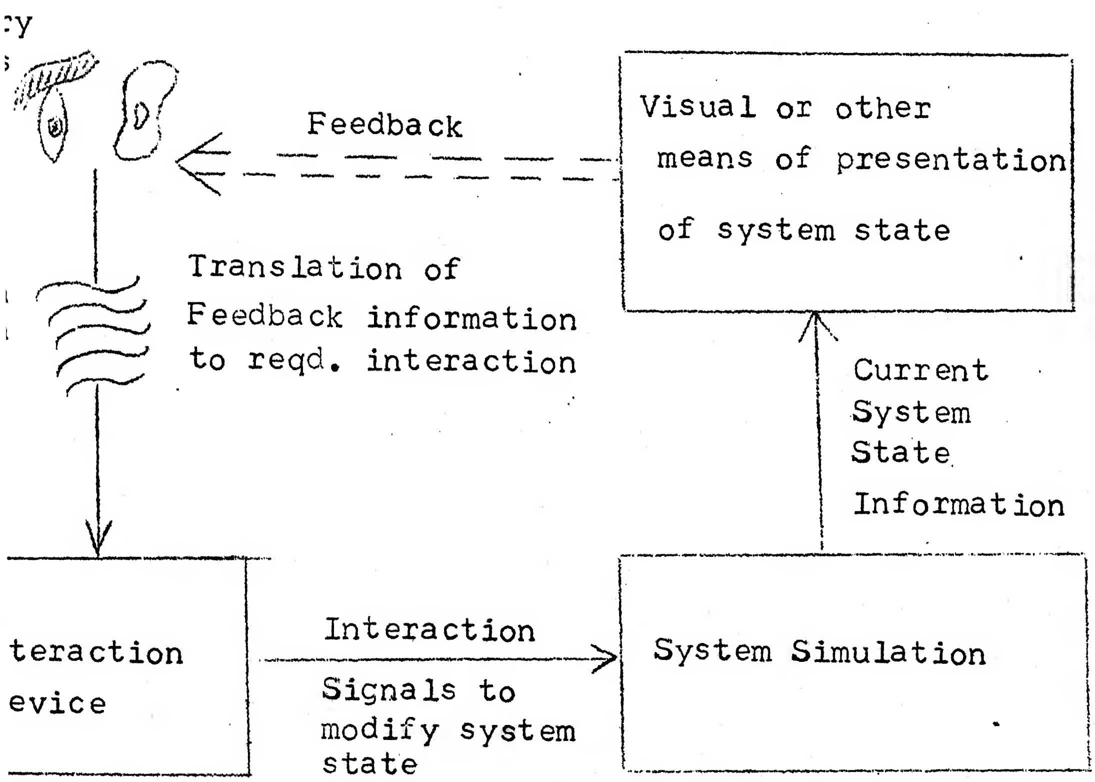


Fig. 2.2: Interactive Simulation: Hardware Configuration

is run, it gives information on system state by some means to the human being, who interprets the information and decides the next course of action. The inputs given by him to the system modify the system state and the system state is conveyed to him again, and the cycle continually repeats.

The potential of interactive simulation as a training medium and for testing military/business strategies is immediately obvious. A novice may sit in an aircraft simulator and learn the flying technique till he is good enough to handle an actual aircraft. Similarly a commander may try out his battle strategies by simulation, to ascertain gain/loss pay-offs associated with each approach, rather than taking an intuitive plunge in the battle field only. If a space-craft develops snags while on mission, the choice out of possible courses of action is decided on earth, by simulating the snag and trying out all possible alternatives.

2.5 COMPUTER GRAPHICS IN SIMULATION

'Seeing is believing', so the saying goes. The best mode of communication is visual, so in an interactive simulation, the feedback information is best provided by visual means. LED displays and other alphanumeric displays have been used, but these provide little information and are inflexible in use i.e. a configuration is generally dedicated to a specific application.

Also, in some applications, the intended visual feed-back is quite complex e.g. a landing aircraft simulator provides synthetic view of moving terrain. Computer graphics serve this function admirably. In general, graphic functions may be implemented on the same computer on which the simulation is run, though additional hardware and software need be incorporated in the system.

2.5.1 Computer Graphics Programming in Interactive Environment:

FOLEY and VANDAM [8] have described the hardware, software and techniques used for implementing interactive computer graphics. A programmer, generally, assumes a graphics package available to him. The graphics program uses facilities provided by the available package. The program creates the following three activities

i) APPLICATION MODEL CONSTRUCTION: The application model stands for a data mass representing the definition of graphic objects.

ii) DESCRIPTION OF OBJECT VIEW: The object is described for calculation of the desired view and display.

iii) INTERACTION HANDLING: The program provides means to convey users interaction to program for specifying modification or changes in the displayed view.

These activities are supplemented by graphic environment control routines available to the programmer.

The first of the three activities provides the basic definition of the graphic entity. It will include definition of size, shape, placement and attributes like colour, pattern etc. It is on the basis of this data that graphic primitives create the object on the device screen.

The second activity creates, transforms and displays the view of the object on the graphic device. This activity may use segment definition and manipulation routines for treating a set of primitives as a single graphic entity.

The third activity consists in providing user interaction by providing communication between I/O devices and the program. A variety of input devices can be used with graphic packages. Logical devices i.e. locators, picks, valuator, button and keyboards can be used. Physically pens, mouse, tablets, joydisk etc. are provided to simulate these logical devices.

The output devices are colour CRT terminals for interactive graphics. One may refer to FOLEY and VANDAM [8] for detailed description of interaction devices and techniques.

2.5.2 Update Dynamics:

The easiest way to present a changed view of a graphic object would be to redraw the altered definition of the view. In real-time graphics simulations, the drawing of complex views will be slow enough to give a break between the previous view and the new one. If animation be the purpose, there will be a noticeable gap between successive frames causing flicker.

This problem is, to some extent, obviated by using segment definition and segment transformation routines provided by a graphics package segments once defined could be rotated, scaled and translated as a single graphics entity. The change of view is comparatively quite fast.

Sometimes this approach, also, may not provide, change of view, fast enough for animation. Then one could draw all possible variations of view as segments and make segments visible and invisible selectively to give effect of smooth motion of the graphic object.

CHAPTER III

FLIGHT AND DAMAGE ASSESSMENT MODELS

This chapter presents flight theory concepts, fuzzy set theory concepts and derivation of flight and damage assessment models.

3.1 THE FLIGHT CONCEPTS

The flight of an aircraft is a complex motion. Six degrees of freedom and numerous interacting variables that effect the aircraft's passage through air, make analysis of it's motion, tedius. For an accurate treatment, in addition concepts of mechanics, aerodynamics, propulsion and meteorology are required to be understood. For the present work, however, a simplified model is attempted.

KERMODE[10] and VONMISFS[11] have explained the theory of flight and the basics involved. Principle of generating lift through flow of stream-lined air around an aerofoil is explained. Following discussion assumes this concept.

An aircraft is made up of a structure called airframe. A pictorial view of an aircraft is presented in Fig. 3.1 to show details of airframe components. An engine is embedded in this structure and generates power to push the aircraft forward. This motion causes stream-lined flow of air to

pass around aerofoil section wings and lift is generated, acting on wings upwards. The lift force, when it counter - balances the weight of the aircraft, causes it to float in the air. The flight of the aircraft continues till forward motion can sustain generation of lift.

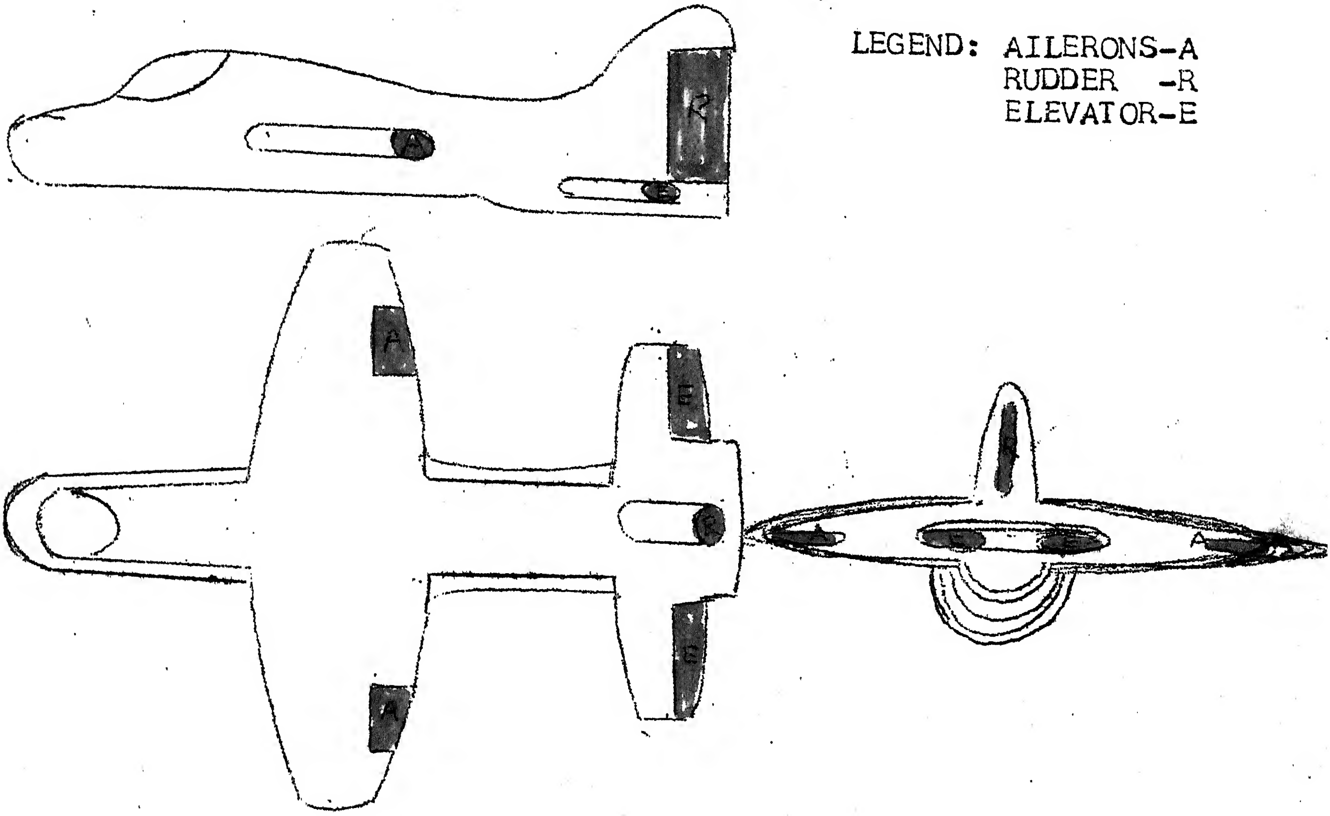


Fig. 3.1: An Aircraft Structure: The Airframe view

During the flight, a pilot manoeuvres the aircraft by moving control surfaces (marked in green in Fig. 3.1) through cockpit-controls and control connections. Consequently, the aircraft is guided on a desired path. There are many other controls/services in the aircraft operations

which enhance efficiency/safety/comfort or serve special purposes. But in this simplified view, we will consider the effect of only attitude controllers i.e. ailerons, elevators and rudders, in addition to throttle. Even these controller have refinements like trim-tabs and horn balancing. We stick to the simplistic view.

3.1.1 Level Flight and Manoeuvers:

As already discussed, the lift generated by forward motion of the aircraft counter-balances the weight of the aircraft. The forward motion of the aircraft is sustained by engine generated 'thrust'. This forward motion is opposed by 'drag' force caused due to inertia and viscosity of the surrounding air.

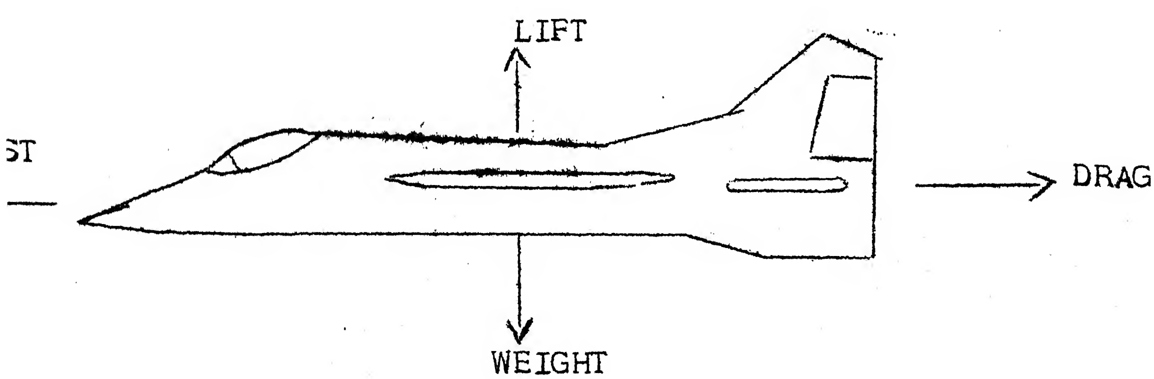


Fig. 3.2: Equilibrium State of Forces: Level Flight

When an aircraft travels at a constant velocity in a horizontal plane, 'lift' equals 'weight' and 'thrust' equals

'drag'. This is called equilibrium state of aerodynamic forces in level flight. These forces act through various points on the airframe. Lift acts through 'centre of pressure' and weight through 'centre of gravity'. Rigging of an aircraft (configuration of airframe elements) ensures that the two points coincide in flight for the equilibrium to take place.

Manoeuvring an aircraft is to alter these four forces in a way to effect requisite change of attitude/ accelerations. Only weight is a quantity over which a pilot has no control (unless dump fuel/load facility is available, but this will be an emergency action even if available). The other parameters are adjusted for the intended manoeuvring.

The manoeuvring of an aircraft can be seen in terms of movements along/around it's three axes, as shown below in Fig. 3.3.

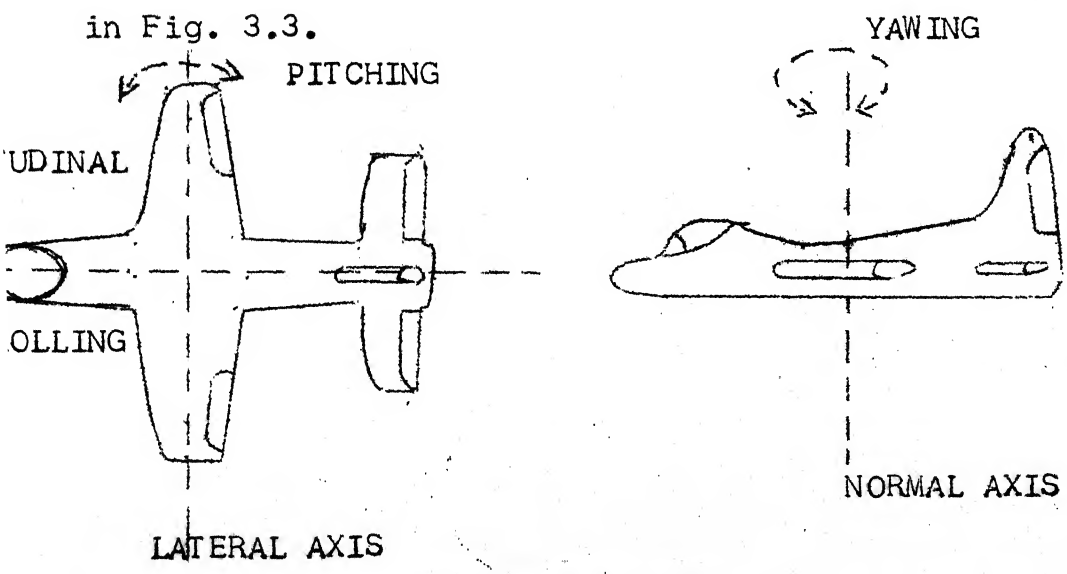


Fig. 3.3: Three Axes of Aircraft

The six degrees of freedom are as follows:

i) Movement along longitudinal axis: forwards or backwards. The forward movement is the natural one. Backwards movement is practically impossible to achieve.

ii) Movement along normal axis: up or down, this is not a natural movement again.

iii) Movement along lateral axis: left or right, or side-slip, it can be a result of rudder application or unsynchronised turn.

iv) Rotation around longitudinal axis: rolling, this movement is the practical control action for turning.

v) Rotation around normal axis: yawing, this movement is the result of a synchronised turn or rudder application

vi) Rotation around lateral axis: pitching, the moment used for climb or descent.

The manoeuvres of an aircraft will be one or combination of the above-listed six movements. In practice, a pilot uses pitching movement for climbing or descending, and rolling movement (co-ordinated by rudder) for turning the aircraft in a desired direction. The movements are obtained by exercising elevators and aileron controls respectively. The rudders are used, in addition, in a turning movement to

During the manoeuvres, there are other control actions, which are applied to offset the effect of differential flow of air around a turning or pitching aircraft; so the combinations of turn and climb/descent make the mathematical modelling of the flight process involved.

3.1.2 Derivation of Simulation Flight Model:

For the sake of simplicity, we consider the rolling and pitching movements independent of each other i.e. we do not incorporate processes like holding-off the bank in a constant rate turn, or changing of hold-off strategy in gliding or climbing turn, in our model. Thus in our model, the total manoeuvre of an aircraft is accumulation of independently analysed roll and pitch movements. We also assume automatic ideal rudder operation. Thus irrespective of pitch attitude, a turn at a particular rate requires banking of the aircraft at a certain angle (with right amount of rudder automatically applied). Following kinds of motion are also ruled out.

i) Side-slip

ii) Tail slide

iii) Up/down slide along normal axis

iv) Yawing

3.1.2.1 Model for Turn:

In a turn, the aircraft moves in a curved path which

is a deviation from it's straight line path. The curved path can be approximated wholly or segmentwise by a circular arc in case of a co-ordinated turn (a turn without side-slip). By Newton's first law of motion, the aircraft tends to maintain it's straight line path. So only an acceleration towards the centre of curve, will force it to follow the circular path.

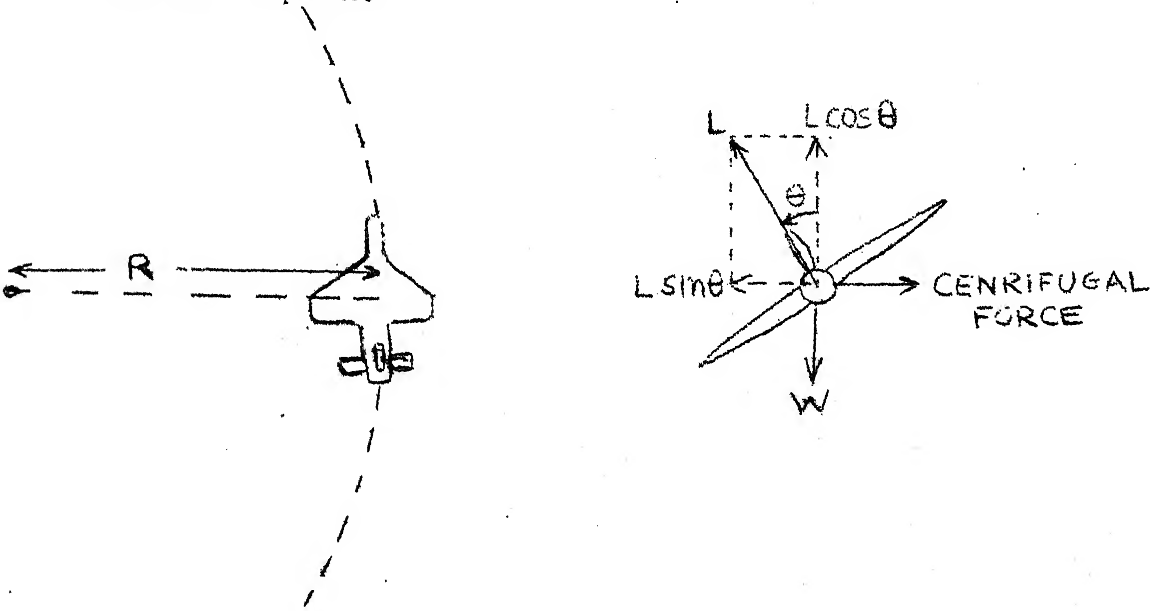


Fig. 3.4: Forces on an Aircraft during Turn

This acceleration is supplied by banking the aircraft. The lift force component in the horizontal plane, then, becomes the centripetal force for aircraft to describe a circle. If an aircraft banks by an angle θ to describe a circle (or section thereof) with radius R, then the lift component L cosθ counter-balances the weight W and horizontal component Lsinθ, which acts as centripetal force, counter

balances the centrifugal force which equals $\frac{W.V^2}{gR}$. Here V is the velocity of the aircraft, g acceleration due to gravity, W the weight and R the radius of the circle. When an aircraft is initiated into a turn, the lift should be sufficiently increased so that subsequent to the bank, the vertical component of lift counter-balances the weight. In practice, a pilot increases the angle of attack or raises throttle setting so as to make a level turn. Otherwise the aircraft will slide down along normal axis. This factor is ignored in the model but is mentioned here for under-scoring the various variables of the 'turn' movement of the aircraft.

To calculate the rate of turn, the procedure can be devised as follows. If an aircraft travels a distance of $2\pi R$ (the circumference of the circle), it turns through 360 degrees. So rate of turn/minute can be given by the following relation

$$\text{rate of turn} = \frac{V.360}{2\pi R.60} \quad \text{deg/min}$$

$$= \frac{3V}{\pi.R} \quad \text{deg} \qquad (3.1)$$

In this relationship R is an unknown variable. But we know that (from Fig. 3.4)

$$\tan\theta = \frac{WV^2}{gR}/W$$

or

$$R = \frac{V^2}{g.\tan\theta} \qquad (3.2)$$

Substituting this value of R from 3.2 into 3.1

$$\text{rate of turn} = \frac{3.V}{\pi.V^2/g \tan\theta} \qquad \text{deg/min}$$

$$= \frac{3\ g \tan\theta}{\pi.V} \qquad \text{deg/min} \qquad (3.3)$$

The speed of an aircraft washes off (decreases) as a result of increased drag which is produced as larger profile is presented to the medium. The rate of speed wash-off depends upon the speed, aircraft type, configuration of aircraft controls at the time of turn etc. The pattern of wash-off, thus, will vary with different types of aircraft.

For the purpose of simulation, a heuristic formula is used to compute this effect. If θ be the angle of bank, the maximum wash-off effect is computed as

$$V_{eff} = V_{init}\ (1\text{-}0.35 \sin\theta) \qquad (3.4)$$

where V_{eff} is the final speed if maximum wash-off is allowed to develope when turn of aircraft is initiated at V_{init} speed by rolling through θ. The formula is representative of actual figures for jet aircraft.

3.1.2.2 Model for Climb/Descent:

At a particular engine power setting, an aircraft has an amount of total energy. The total energy can be thought of

as a sum of potential energy (altitude) and kinetic energy (because of forward momentum). Keeping engine power constant and without change in configuration (change of drag variable by exercising lift augmenters etc.), one form (potential energy) can be increased at the expense of other (kinetic energy) and vice versa. Increase of engine power adds to total energy, which can result as increase of either component or both. A pilot adjusts the engine power setting (throttle) and pitch attitude of the aircraft to achieve a desired combination of aircraft speed and climb/descent rates.

Once again, the pitching movement is considered in isolation from other component manoeuvres for the purpose of modelling. The inaccuracies of such an assumption have already been discussed for 'modelling the turn'. Simplification is necessitated to keep the model from being cumbersome. The formulae adopted is heuristic but representative for the range of pitch attitude variation modelled for this simulation.

$$V_{eff} = V_{init} (1 - \sin \emptyset) \qquad (3.5)$$

where V_{eff} is the maximum equilibrium speed achieved if the aircraft pitches by angle $\emptyset$, when travelling at a steady speed of V_{init}. No distinction is made between the simple pitching or pitching during a turn in this model.

With these models, we can compute the aircraft path during level flight as well as manoeuvres.

3.2 FUZZY SET THEORY AND FUZZY MODELLING

3.2.1 Need for Fuzzy set Theoretic Approach:

The theory of fuzzy sets is a step towards a reproachment between the precision of the classical mathematics and pervasive imprecision of the real world. A computer operates in strict conformity with mathematical logic but cannot think and interpolate information in imprecise, non-quantitative or fuzzy terms like a human being can. This novel ability of a human mind enables one to decipher sloppy handwriting, focus on pertinent information in larger text, summarise information, include additional context on imprecise information etc.

Computer solutions of these real world problems have, for a long time, used the classical approach of reducing the fuzzy quantities, fuzzy relationships and fuzzy rules of inference to classical set theory models, which have binary logic as its basis. One tries to obviate the uncertainties by using probabilistic techniques. But inherent limitations of the classical methods are due to approximations of a fuzzy entity by an equivalent binary entity in such a problem.

Classical methods, despite their limitations, have sufficed for numerous applications, but in the fields of Artificial intelligence, and applications to soft sciences like management, linguistics, economics, psychology and biology, the limitations of classical approach are accentuated. Fuzzy set theoretic approach is the need for solution of these problems.

Solution of problems by fuzzy set theoretic approach have developed in diverse directions three of which are

i) Development of special software to run binary logic machines: Autoadaptive programs are such an effort in the field of artificial intelligence.

ii) Development of a new multivalued logic. POST (1921) LUKASIEWICZ (1937) and MOISIL (1940) as mentioned by KAUFMANN [13], have in different presentations, pointed towards this development. All these authors have presented general theories for construction of such η-ary logic.

iii) Development of information processing machines other than computers, which treat information globally, without necessarily passing through a sequential treatment. These machines may be called combiners or parallel processors, and treat fuzziness by Fuzzy logic itself rather than binary logic.

As is evident, the later two fields are in early developmental stages. The first development finds application in a variety of fields. Many of these applications are described in papers, some of which can be found listed by KANDEL and LEE[12] and KAUFMANN[13].

In the present work, it is intended to solve the weapon interaction problem of the 'air combat' by fuzzy set theoretic software formulation rather than probabilistic techniques. Before defining the problem and deriving the fuzzy model, it may be in order to present the basics of 'Fuzzy set theory'. A brief coverage of the theory follows hence

3.2.2 Fuzzy Set Theory:

The conventional or the abstract set is defined as a collection of objects, in which nothing special is assumed about the nature or properties of the individual elements. In an object space, an element may or may not belong to a set. In order to indicate this membership, the concept of characteristic function may be used e.g.

$$\mu_A(x) = 1 \quad \text{if} \quad x \in A$$
$$= 0 \quad \text{if} \quad x \notin A$$

A characteristic function takes only binary values i.e. 0/1. The object is present or absent, white or black, true

or false. Any in-between situation can be resolved by probabilistic techniques, such as rolling a dice, monte-carlo method etc. to determine whether the in-between situation could be designated 0 or 1. The real world, however, involves constructs, which are not sets in the classical sense, but rather fuzzy sets i.e. classes with unsharp boundaries, in which the transition from membership to non-membership is gradual rather than abrupt e.g. black and white are interspersed with various hues of gray in-between.

The fuzzy set theory, to represent the gradual change, introduces the concept of weighted membership. We could now represent values of subjective terms like nearer, more or less on time, etc. The subjective and comparative relationships of the real world can, thus, be translated into a graded membership function which, instead of assuming only 0/1 values, assumes a continuum of values in the closed interval [0,1].

Following is an example to explain the concept. Suppose one defines a set, having as members, numbers much greater than 1. Is 25 a member of this set ?. Is 100, too, a member? What is the distinction between the membership of 25 and 100?

The answer to the above questions can be given as per 'fuzzy set theory' principles. The answer could be a

fuzzy set which has 25 as it's member with, say, a membership grade 0.5 and 100 with a grade of membership 1. The assignment of membership grade is arbitrary in the above example. It could be determined by a specified relationship in other cases.

The continuum could be quantized. A discretised set of grades could be the domain of the membership function as well. The important thing is that membership function can assume values other than 0 and 1. Such a discrete multi-valued fuzzy logic could find implementation in the n-ary logic hardware, as conceived by POST, LUKASIEWICZ and MOISIL. KANDEL and LEE[12] have defined fuzzy sets and their characteristics. KAUFMANN[13] has described various operations on these sets and interaction between fuzzy variables in terms of fuzzy graphs and relations. KAUFMANN [13]also describes various membership functions, two of which are adopted for damage assessment formulation in this work.

3.2.3 Fuzzy Models and Probabilistic Models:

Fuzzy set concept is a superset of conventional set concept. If one were to allow only two values out of continuum of closed [0,1] interval, a fuzzy set reduces to a conventional set. If one uses probabilistic techniques to map in-between values to 0 or 1, the model is called a probabilistic model.

A probability measure itself may be fuzzy i.e. uncertain e.g. one could say that the probability of getting a head on tossing a coin is 0.5 (with a possibility of 1) while as the probability of Marxistswinning the next assembly elections in West Bengal may not be expressed in exact probability figures. One may, though, say that the probability is 0.6 with a possibility of 0.8, 0.2 possibility pointing towards swaying of the electorate by Prime Minister's offensive compaigning. Fuzzy concepts, then, could be applied to solve uncertainity of probability itself and such a probability is known as fuzzy probability.

But in essence, a fuzzy model provides solutions by possibility theory methods rather than probability theory methods and multivalued/continuum of representation of fuzzy quantities is possible in fuzzy models.

3.2.4 Probability Measure of Fuzzy Events:

The basic notions in probability theory are that of an event and it's probability. An event A is a subset of sample space S and it's probability is it's mapping from sample space to a real number in closed interval [0,1].

If the event A be a fuzzy event, rather than a conventional sub-set of sample space, we would still like to have it's probability measure. This would be a

requirement in solving many real world problems. Such an extention can significantly enlarge the domain of applicability of probability theory, specially in those fields, where fuzziness is a pervasive phenomenon. The following describes such extention of probability theory by an example.

Let us assume that a fair die is thrown. Let A be a fuzzy event which denotes that the number that turns up is close to 3. The problem is to find out the probability of event A.

To solve the above problem, it is essential to assign membership functions to various possible outcomes.

$$f_A(x) \quad f_A(1) = 0.4,\ f_A(2) = 0.6,\ f_A(3) = 1.0$$

$$f_A(4) = 0.6,\ f_A(5) = 0.4,\ f_A(6) = 0.2$$

Then the probability measure of the fuzzy event A can be calculated as follows

$$P(A) = \sum_{i=1}^{6} p(x_i) \cdot f_A(x_i)$$

Assuming a fair die $p(x_i) = \frac{1}{6}$ for $i = 1, 6$.

$$P(A) = \frac{1}{6} \times 0.4 + \frac{1}{6} \times 0.6 + \frac{1}{6} \times 1.0 + \frac{1}{6} \times 0.6 +$$

$$\frac{1}{6} \times 0.4 + \frac{1}{6} \times 0.2 = 0.533$$

The probability of a fuzzy event is the expected value of it's membership function calculated over the domain of fuzzy set.

The above said principles can be applied to the problem at hand, in this work.

3.2.5 The Damage Assessment Problem:

The problem to be solved in the present work is that of damage assessment. In that, there are two aircraft in flight in a pre-defined combat space, involved in a dogfight. One fires a weapon at the other at a particular moment. The present position (at the time of weapon release), the velocity vectors of both the aircraft and weapon, are known. For the sake of simplicity of simulation, the effect of the following factors are ignored.

i) Aircraft manoeuvres during the time of the flight of weapon

ii) Any interfering aerodynamic forces e.g. cross-wind, turbulence etc.

iii) Gravitational effects on weapon flight.

It is intended to compute interaction between the weapon and the target aircraft. The classical approach in the solution of this problem would have been to compute the

minimum distance between the target aircraft and weapon during the time of flight of the weapon, assign a probabilistic distribution to the chances of kill with respect to the minimum distance and then use monte-carlo technique to determine whether a kill has been scored. The answer would be a kill/escape.

But in real actions (air combats) it is well known that the extent of damage is a graded value rather than a pure escape/kill. As a weapon hits closer to the centre of an aircraft, it is likely to inflict more damage. One could subjectively argue that 5 m square loss of tail-plane is costlier than same area lost in wings, but a simplistic argument could be given, in favour, too.

Also an aircraft can take a certain amount of 'beating'. It is possible for the aircraft to sustain damage and still be flyworthy. As the aircraft sustains and accumulates more of hits/damage, it's flying characteristics deteriorate till it can fly no longer. For the partial damage assessment, a graded membership function is assigned to represent damage inflicted. The function correlates the minimum distance between the weapon and target, during weapon flight, to the damage inflicted. The fuzzy variable translates to partial damage, which in accumulated manner, is used to assess aircraft condition.

3.2.6 Damage Assessment Models:

Models for damage assessments for anti-aircraft weapons, as well as bomb-drops can now be devised. The assumptions made in case of anti-aircraft weapons i.e. missile and guns are

i) The weapons are fired towards the target, which means that the distance between the weapon and target reduces after the weapon is fired.

ii) The minimum skew distance between the two vectors representing weapon and target velocities can be computed.

iii) The distance required to be travelled by the weapon subsequent to the launch to achieve minimum skew distance of (ii) can also be computed.

Assessment of damage caused by a bomb to a ground installation assumes the following:

i) Bombs are free-fall type

ii) Simultaneous release causes bombs to hit the same point on surface.

In addition the following is assumed for all weapons

i) There are no 'duds'. All weapons explode as and when expected to.

ii) There are no extraneous factors like wind, gravity effects affecting the interaction.

3.2.6.1 Missile Damage Assessment Model:

The damage assessment follows the criteria enumerated below

i) A direct hit ($d_{min} = 0$) inflicts full damage

ii) Terminal, guidance corrects aim deviations within this cone is as good as 'dead on target'.

iii) Beyond this cone, a missile cannot guide itself so is wasted. On surface of cone, the damage dimunishes quickly.

The angle calculation is substituted by 'ratio of distance' calculations. The independent variable is $\frac{d_{min}}{x}$ where d_{min} is the minimum skew distance while as x is the distance travelled by the weapon to position itself closest to the target. The membership function is graphically shown in Fig. 3.5 and the mathematical formulation of the model is as follows.

$$d_q = 0 \quad \text{for} \quad \frac{d_{min}}{x} > 0.15$$

$$= 20(0.15 - \frac{d_{min}}{x}) \cdot K_1 \text{ for } 0.1 < \frac{d_{min}}{x} < 0.15$$

$$= K_1 \quad \text{for} \quad \frac{d_{min}}{x} < 0.1$$

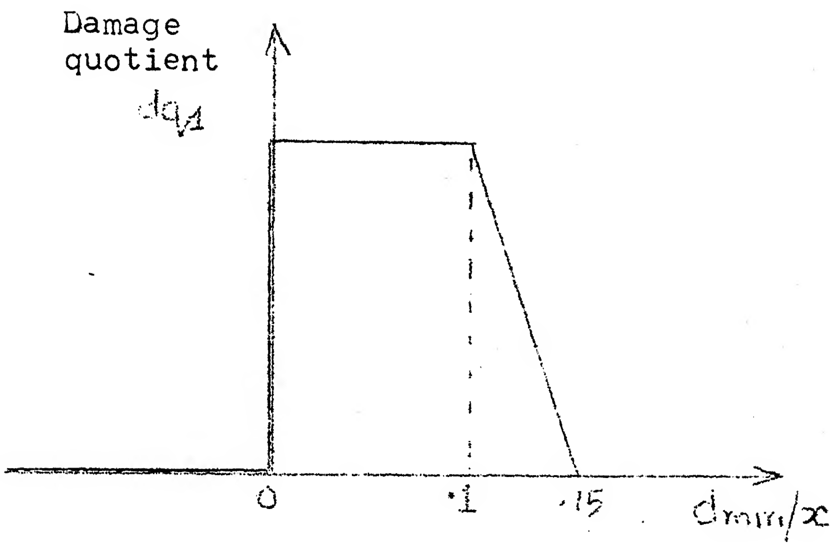


Fig. 3.5: Missile Damage Membership Function

3.2.6.2 Gunfire Damage Assessment Model:

The assessment follows criteria given below:

i) Maximum damage is inflicted if gun fires 'dead on target'.

ii) Damage decreases as aim wanders off-target

iii) Beyond a limit, gunfire off-target is wasted.

The membership function is shown below and formulation of damage assessment follows Figure 3.6.

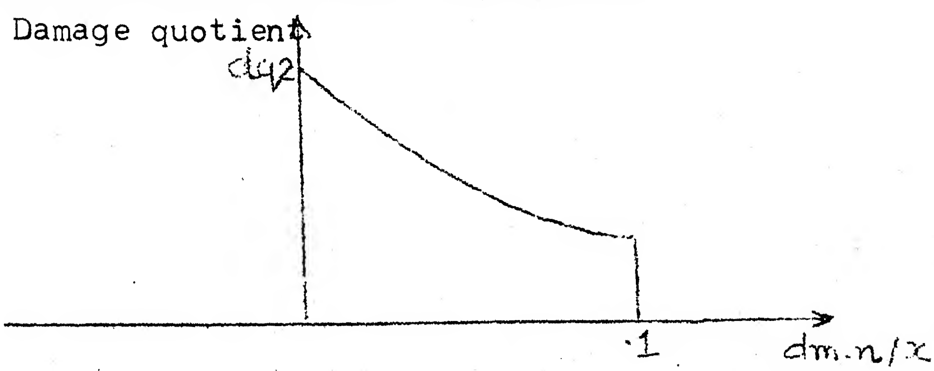


Fig. 3.6: Gunfire Damage Membership Function.

$$d_{q2} = 0 \text{ for } \quad \frac{d_{min}}{x} > 0.1$$

$$= e^{-C\frac{d_{min}}{x}} \qquad 0 < \frac{d_{min}}{x} < 0.1$$

e should be chosen > 10 for the intended fall-off of damage with wandering off of aim.

3.2.6.3 Bomb Damage Assessment Model:

This model differs from the missile and gun damage modules. Here the distance off target is the only parameter of consequence, as the target is stationary. The criteria are as given below.

i) A direct hit destroys the target

ii) A hit, slightly off target, but still within it's effective zone, also destroys target

iii) The damage reduces as off-target distance increases

iv) Beyond 500 meters, bomb causes no damage.

The membership function is graphically represented by Fig. 3.7.

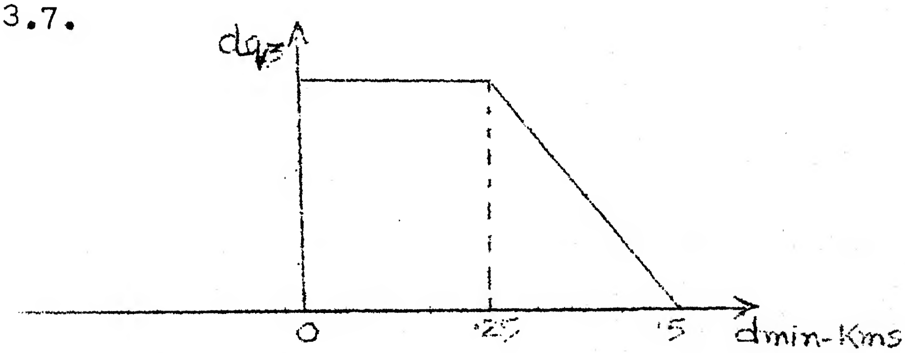


Fig. 3.7: Bomb Damage Membership Function

The assessment is effected by the following formulation.

$$d_{q3} = K_3 \quad \text{for } d_{min} < 0.25$$

$$= 4(.5 - d_{min})\, K_3 \text{ for } 0.25 < d_{min} < 0.5$$

$$= 0 \quad \text{for } d_{min} > 0.5$$

These models can be incorporated in the system design for weapon interaction result calculations.

CHAPTER IV

SYSTEM DESIGN

The simulation procedure on the basis of models outlined in the last chapter can now be described. The procedure includes:

a) Description of the simulation run

b) Block diagram of the implementation

c) Detailed specifications of the game parameters

d) Flow chart of the program

e) Other implementation details

4.1 DESCRIPTION OF THE SIMULATION RUN

The program run begins with initialisation of the parameters and graphic view and screen feedback of instructions to the users on both the terminals. Each user is explained the simulation and is prompted to initiate the simulation run by pressing a particular key. On appropriate inter-action, the attacker initiates the simulation run.

The attacker is positioned at (0,0) corner of the 400 km x 400 km enemy area, with an initial level speed of 750 kms proceeding in 045^{o} direction at 3 kms altitude. The control of the aircraft is now with the attacker console user.

The defender, at the start of simulation is positioned, at rest, on ground at (200,200) point i.e. at the base, which he is required to defend. The defender gets information about attacker's position through radar view in his cockpit and can choose to take-off any time (by pressing a key i.e. T). On taking-off, the defender is positioned at point (200,200) proceeding with a level speed of 750 km/h in 00^{o} direction at 1 km altitude.

The simulation proceeds, with both users trying to achieve their objectives without going out of bounds (which is the designated 400 km x 400 km x 10 km volume) and within their respective endurance limits. The simulation terminates when either of the users does one or more of the following

a) achieves his objective i.e. successful bombing or interception

b) goes out of bounds

c) uses up all his endurance

Partial game results may be calculated for an 'end of endurance' termination. The simulation ends by giving result messages to either of the users.

4.2 BLOCK DESCRIPTION OF IMPLEMENTATION

The computer implementation of simulation consists of programs to effect the following functions of the simulation.

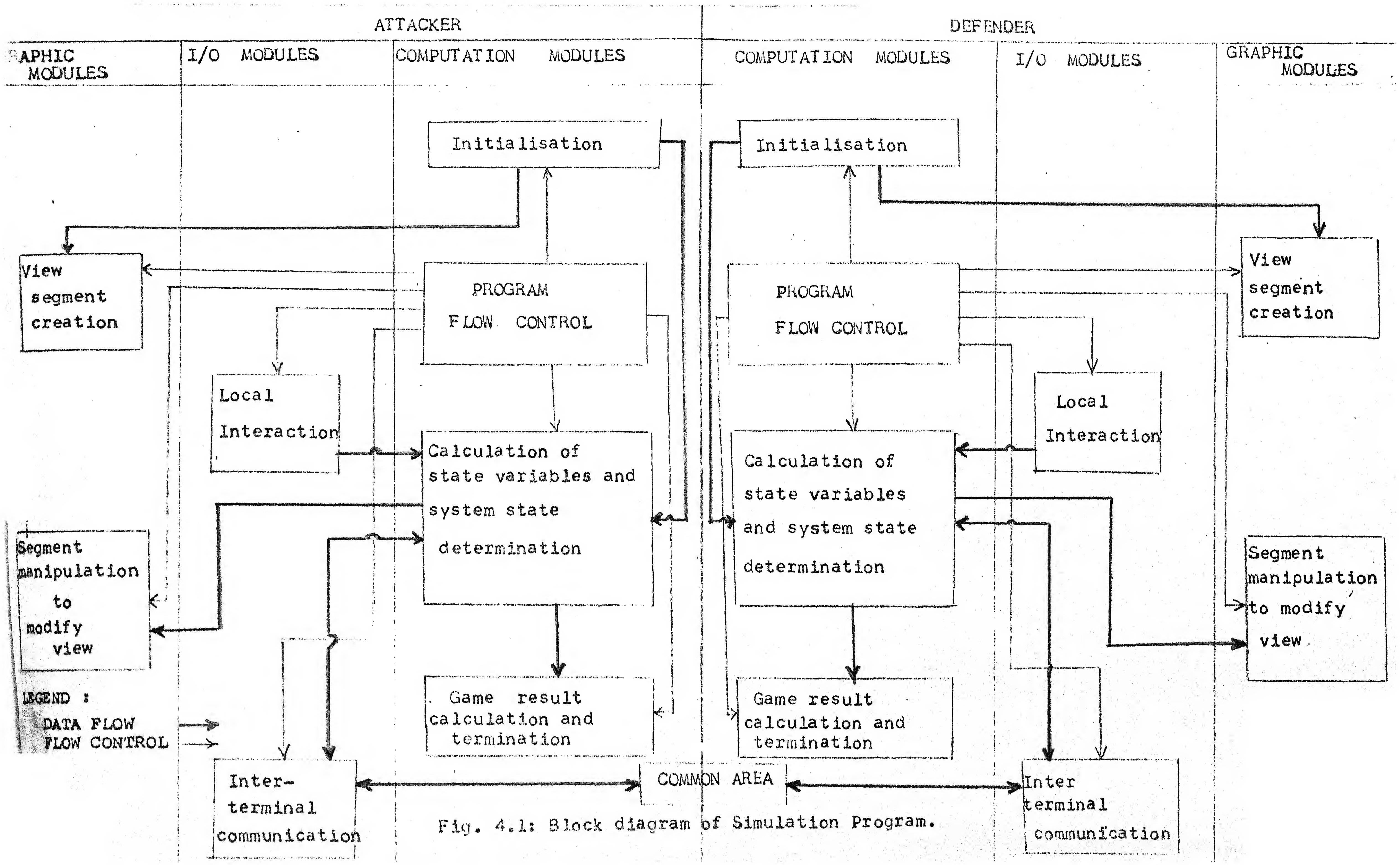


Fig. 4.1: Block diagram of Simulation Program.

a) Initialisation and initial view generation

b) Interaction by users and inter-terminal communication (INPUT/OUTPUT)

c) Calculation of state variable values and system state determination (COMPUTATION)

d) GRAPHIC representation of instantaneous system state (OUTPUT)

e) Computation of game results (TERMINATION)

The implementation is organised as shown in the block diagram in Fig. 4.1 to achieve the above mentioned functions. The block diagram also shows the data flow and control flow between various modules of the implementation. The modules are shown to be segregated function-wise. Technique-wise, two similar modules may use same software methods for implementation. The software techniques are briefly described in Section 4.5.

The block diagram of the simulation is self-explanatory. A brief explanation however follows. On each terminal, an independent program runs to create an initial system state and it's graphic representation. The system state representation on two terminals is synchronised by prompting the users to generate initiating signals and then interchanging information subsequent to the users initiating the simulation

run. Once synchronised time/position information about other aircraft is available, the total system state can be determined and graphically represented. The state variables are updated by taking inter-action into consideration and the other aircraft information is renewed once in each computation cycle.

Each time, a new system state is calculated, the changed view is presented and in case of result variables attaining a specified value, the termination is initiated. In that, the game results are calculated and requisite messages displayed. Subsequent to this, the simulation is terminated.

The inter-terminal communication takes place by read/write operations by either terminal program in a common file. The access by two programs is asynchronous and is expected to be random enough that a simultaneous access is highly improbable. In case a simultaneous access does take place, the program opening the file first retains access whereas the other skips read/write for that computation cycle. A computation cycle is short enough not to effect system performance by skip of one cycle.

4.3 DETAILED SPECIFICATIONS OF GAME PARAMETERS

4.3.1 General Specifications:

a) Bounds of operations for both aircraft is $400 \times 400 \times 10\ km^3$ volume

b) Game termination is the earliest of the three events namely end of endurance, achievement of task by either user and out of bounds travel of either aircraft

c) The target area (base-T) is a 3 km circular area around point (200,200).

4.3.2 Specifications of Attacker's Parameters:

a) No. of Bombs - 4

b) No. of missiles (air to air) - 2

c) Air to air gun munition - 4 Bursts

d) Max. level speed - 1200 km/h

e) Min. speed - 450 km/h

f) Max. speed - 1600 km/h

g) Max. ceiling - 10 kms

h) Endurance with full stores - 15 minutes

j) The correspondence between the throttle setting and maximum level speed attainable is as described in Table 4.1.

Table 4.1: Throttle setting/Max. level speed relationships for attacker aircraft.

S.No.	Throttle Setting	Max. Level Speed
1	60%	600 kmph
2	70%	750 kmps
3	80%	950 kmph
4	90%	1100 kmph
5	100%	1200 kmps

4.3.3 Specification of Defender's Parameters:

a) No. of air to air missiles - 3

b) Air to air gun munition - 8 Bursts

c) Max. level speed - 1600 km/h

d) Min. speed - 600 km/h

e) Max. speed - 2000 km/h

f) Max. ceiling - 20 kms

g) Endurance with full stores - 5 minutes

h) The throttle setting and maximum attainable level speed relationship is described by Table 4.2.

Table 4.2: Throttle setting/max. level speed relationship for defender aircraft.

S.No.	Throttle setting	Max. level speed
1	60%	700 kmph
2	70%	950 kmph
3	80%	1300 kmph
4	90%	1500 kmph
5	100%	1600 kmph

4.4 FLOW CHART OF THE PROGRAM

The flow chart is shown in Fig. 4.2. For clarity of presentation, the activities have been segregated into graphic, interaction and computation activities. (The classifications do not follow the rigorous definitions of the terms representing the class). Each flow chart activity translates into one or more subroutines or segments of the program.

The flow of the simulation can be traced from the flow-chart. At the start, setting up of system environment and initialisation of state-variables is followed by initial view as well as variable segments generation. Then the

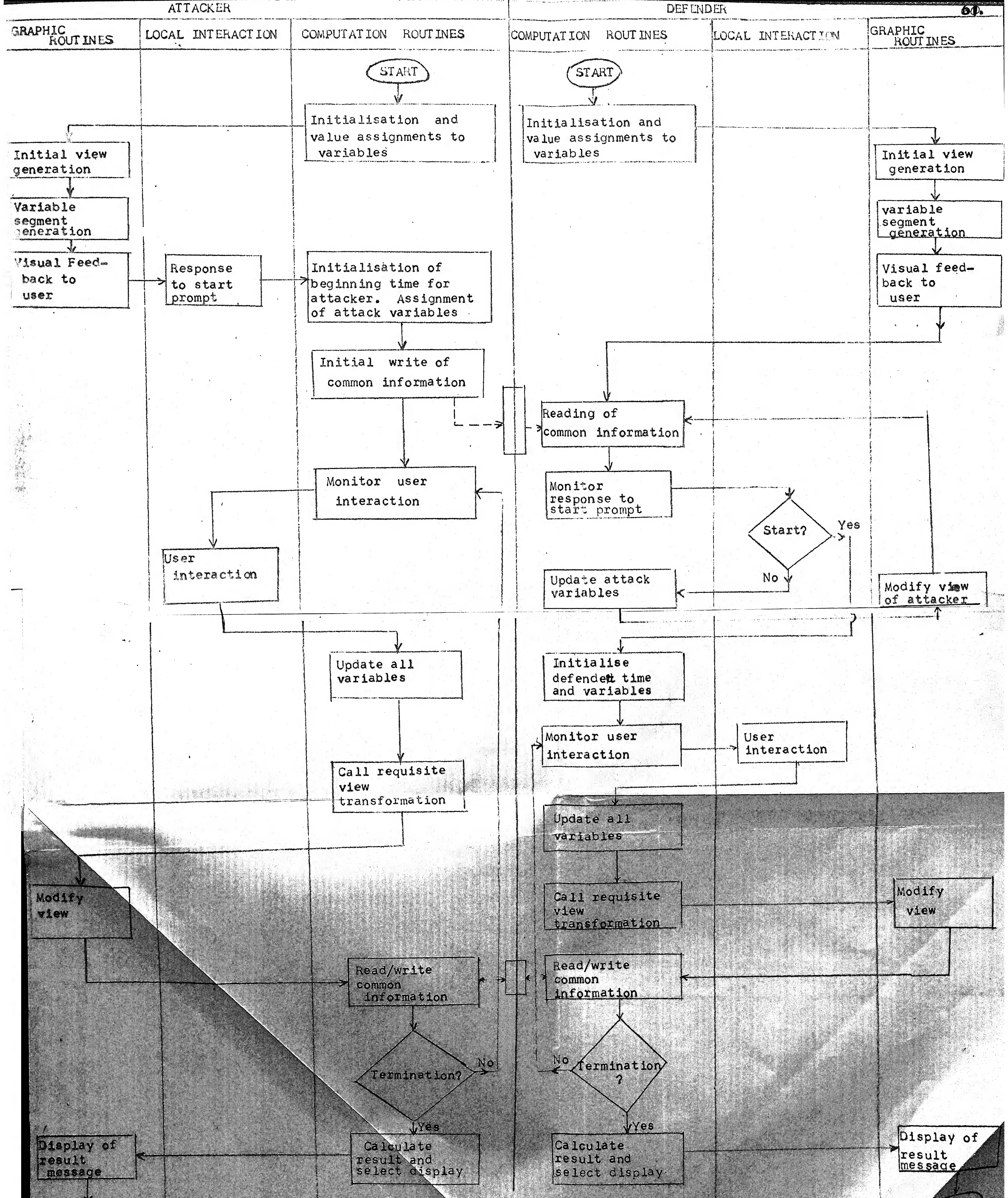


Fig. 4.2: Flow chart of Simulation Program

screen feedback to the user instructs him how to use the simulation. The user is prompted to initiate the simulation run. Subsequent to the desired interaction, the simulation run begins. The two users can now effect interaction with simulation and change state variables to put system in a desired state and achieve their designated tasks. The simulation continues till any of the conditions for termination of simulation is fulfilled. Then, the simulation exits the interactive loop, calculates and announces (graphically) the game results and terminates.

4.5 OTHER IMPLEMENTATION DETAILS:

This simulation is implemented on a NORSK-DATA 560/CX computer in NORSK-DATA super set of FORTRAN-77 using PLOT-10, GKS graphics package to give graphic presentation on two TEKTRONIX-4109A terminals. The program development and also execution is effected on SINTRAN-III operating system and attendent utilities have assisted in program development. A brief write-up, as follows, describes the techniques adopted for specific program module's implementation.

4.5.1 Graphic Routines:

The simulated cockpit view is generated by making calls to PLOT-10 GKS library routines. The change of views is effected by firstly storing all possible representations

of various state variables as invisible segments and then making them selectively visible to represent the instant state variable values. NORSK-DATA PLOT-10/GKS manual [14] may be referred to for description of graphic environment routines, primitives routines, attributes setting routines etc. as provided by ND PLOT-10 GKS implementation.

4.5.2 Input Routines:

The interaction input is monitored by scanning the keyboard buffer once in each computation cycle. To keep interaction from being swamped by a single key repetition, the keyboard buffer is flushed after each read.

The inter-terminal communication is effected by reading/writing variable values in a common file. A terminal program reads the other terminal variable values and writes it's own ideally once per computation cycle. The rate may decrease if simultaneous access occurs. The monitor calls used for both the above-mentioned I/O functions, are described in NORSK DATA manuals namely, Time sharing/ Batch guide [15], SINTRAN III reference manual [16].

4.5.3 Computation Routines:

These routines operate on data and interaction values to calculate the consequent system state. In addition, time keeping functions, weapon interaction solution and game results calculations are also performed. These routines use the NORSK-DATA superset of FORTRAN-77 as described in ND-FORTRAN-77 manual [17]. These routines use models devised in Chapter 3.

CHAPTER V

CONCLUSIONS AND SUGGESTIONS FOR FURTHER WORK

5.1 CONCLUSIONS

This work indicates the possibility of interactive air combat simulation implementation by software techniques. A conventional simulator using dedicated hardware and soft ware can not be equlled in terms of providing real-life control handling and acceleration effects. But a software implementation is much more cost-effective, flexible and portable.

The present work is an attempt towards the goal of getting a proven simulation package. Due to limited resources during the course-work, it has been possible to develop the package to a limit of sophistication. The package may be brought up to the 'general usage' level by further work as suggested later.

The present work uses scan of activities approach for its input and computation functions. Interrupt mode input could be employed with consequent gain in execution speed but have not been tried due to paucity of time.

The graphic view transformation has been implemented by selective visibility of all view combinations. The animation of the view is satisfactory but the time taken to pre-generate all variable-visibility segments is upward of 30 minutes for this implementation. A reduction, in this set up time is effected by generating the blip-position segments in execution run only.

The delay in computation or activity scan overheads been allowed to represent inertia of display. A time-study for stochastic variations of invoking and execution time of each module could be used to model inertia to conform to pre-defined standards, but has not been attempted.

The implementation has a low-level of error-recovery feature. The user feed back of warning messages is implemented to a limited extent.

5.2 SUGGESTION FOR FURTHER WORK

This simulation could be used as a starting model for a more sophisticated model. The suggested updating, in descending order of priority is as follows:

i) The local interaction as well as inter-terminal communication can be made interrupt based. This will avoid overheads for scanning and monitoring inputs.

ii) The aircraft flight modules could be improved to include all flight profiles.

iii) The damage assessment modules could operate on varying aircraft parameters rather than those at the time of weapon launch.

iv) The graphic routines could be implemented in lower-level primitives to improve display speeds.

v) The information available on attackers behaviour could be stochastically modelled to represent real-life situations.

vi) The models could be expanded to cover multi-aircraft scenarios.

vii) Extention of work to include dedicated hardware to develop operational military systems.

APPENDIX A

GLOSSARY OF TERMS

Air combat - an engagement between two or many aircraft, in which, the adversary try to shoot down or drive away each other.

Air defence - usage of aircraft and other weapons to deny the enemy air strikes over friendly area

Air-strike - Usage of aircraft to fire weapons/drop bombs on enemy ground targets.

Air superiority - A state in which enemy is unable to mount air-missions.

Battle - A short engagement between two adversary forces.

Ceiling - Maximum attainable altitude

Circuit - A pre-determined pattern of flight around an airfield.

Close air support - Offensive air mission to assist ground forces.

Cockpit - Portion of the aircraft to accommodate the pilot. All control actions originate from the cockpit.

Defender - One whose task is to stop the enemy from mounting an offensive mission.

Dog fight - An engagement between two adversary aircraft, in which each tries to position himself behind the other to effectively fire the aircombat weapons.

ECM - Electronic counter measures. Emitting electronic signals to gain or misguide enemy's electronic sensors.

ECCM - Measure to counter enemy's ECM

Endurance - Fuel quantity described in terms of time for which the aircraft can stay at loat.

EW - Electronic warfare, term used to encompass ECM, ECCM and electronic surveillance.

Fighter - An aircraft which specialises in air-combat and superiority role.

Fighter-bomber - An aircraft used to launch offensive missions on close-air-support or interdiction unescorted. It has capabilities to engage in air-combat if intercepted by enemy aircraft.

Flight-envelope - A virtual curve along which the flight of an aircraft takes place from launch to recovery

Interceptor - An aircraft which engages an incoming enemy aircraft in aircombat.

Interdiction - Air mission against enemy's supply-lines.

Level speed - Speed of aircraft travelling in a horizontal plane.

Manoeuvre - Change of aircraft speed vector along one or more of the three axes by changing of control configuration of the airframe.

Munition - A war head which can be fired by a weapon.

Sortie - An offensive or defensive mission.

Stores - All weapons on board an aircraft.

Strategy - A pre-formulated plan for conduct of a battle or war.

Surveillance - Monitoring visually or by electronic means, enemy's state of battle positions.

Tactics - Pre-planned responses to enemy manoeuvres

Transport-support - Use of transport aircraft for mobility of ground forces and weapon systems.

War - A confruntation between two countries, in which both use their military forces to impose their decisions on the other.

OK>

OK>

OK>

OK>

TABLE B.1:OUTPUT DATA

FROM GUN DAMAGE SIMULATION RUN

MIN_DIST	GUNDAM
.000092	599.45
.007904	554.40
.015900	511.80
.023895	472.47
.031921	436.03
.039917	402.53
.047913	371.59
.055908	343.04
.063904	316.68
.071899	292.35
.079895	269.88
.087921	249.07
.095917	229.93
.103912	.00
.111908	.00
.119904	.00
.127899	.00
.135895	.00
.143921	.00
.151917	.00
.159912	.00
.167908	.00
.175903	.00
.183899	.00
.191895	.00

STOP 0
OK>

OK>

OK>

OK>

OK>

OK>

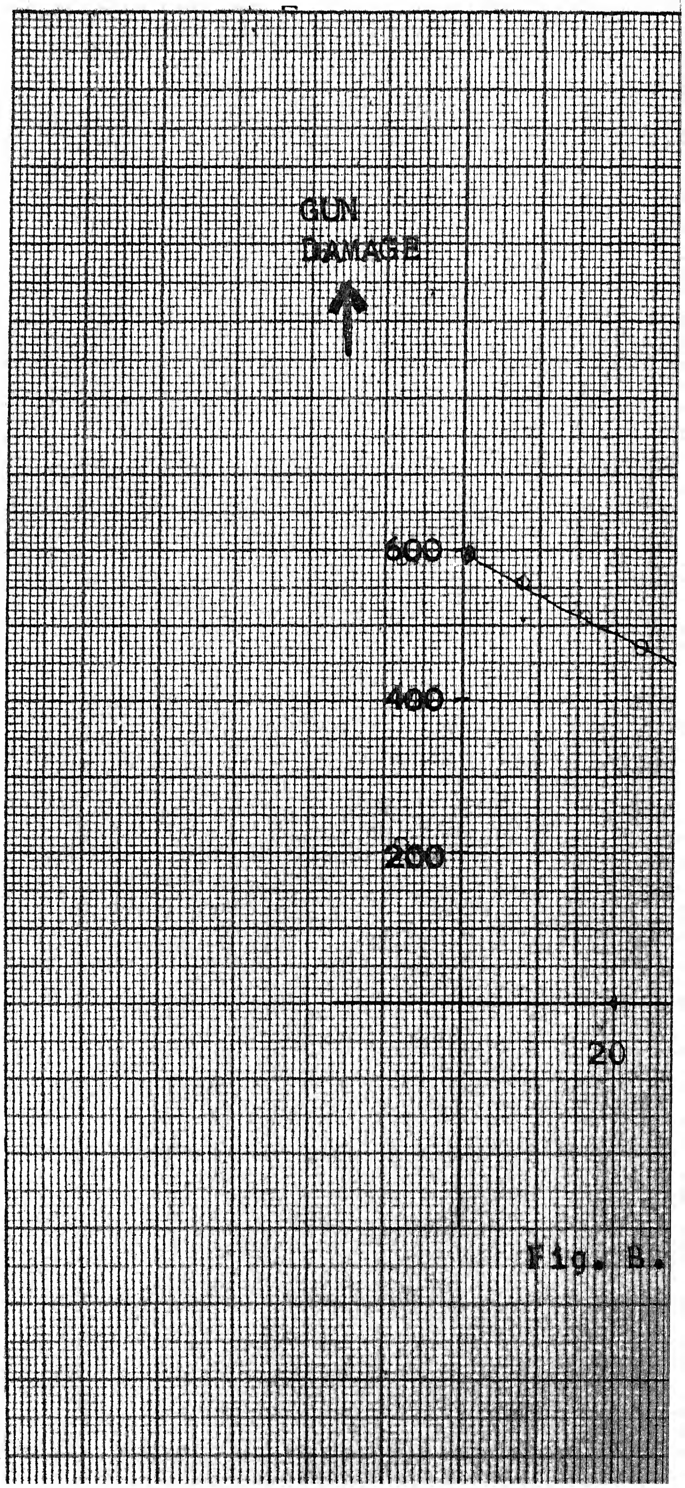
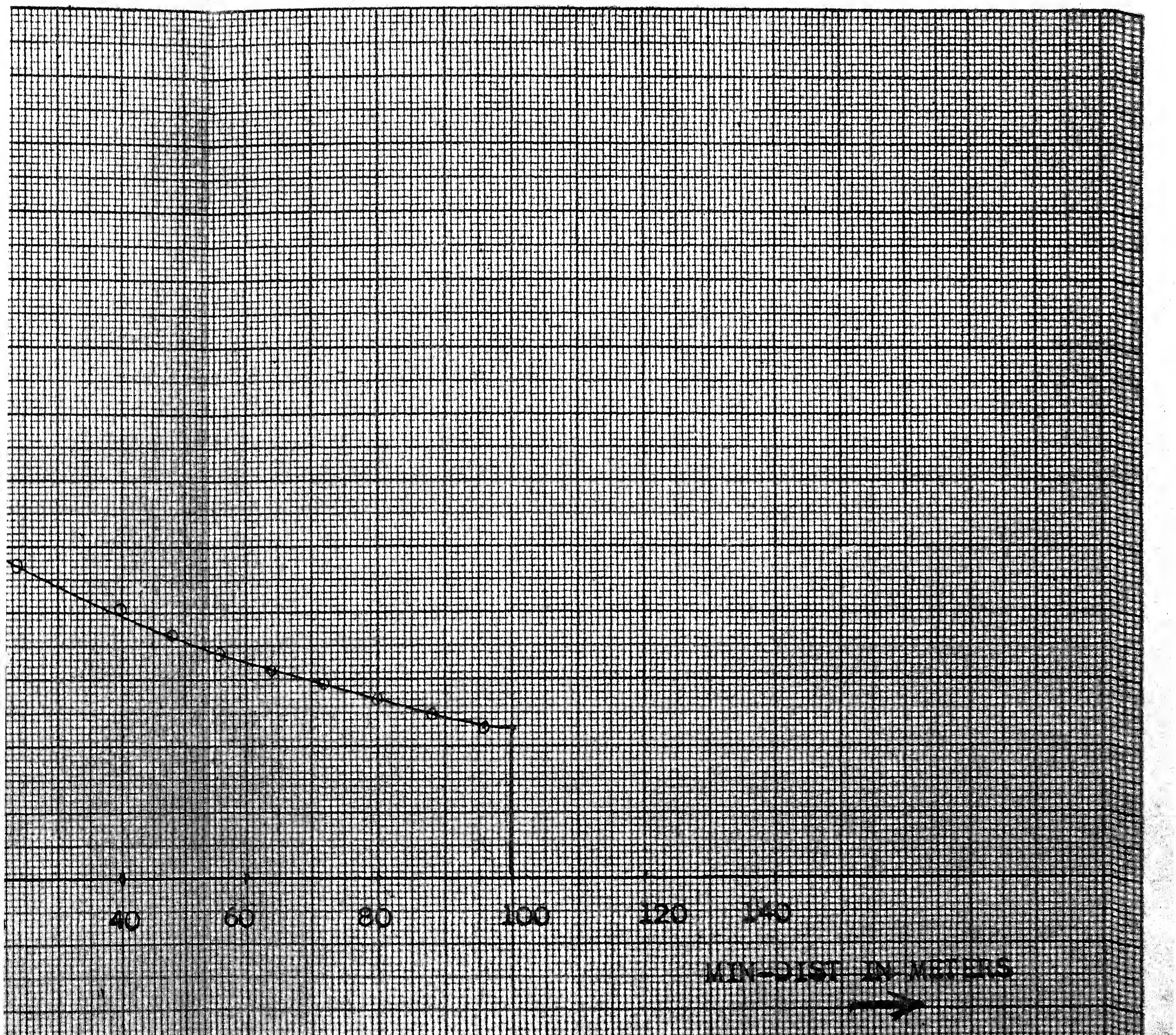


Fig. B.



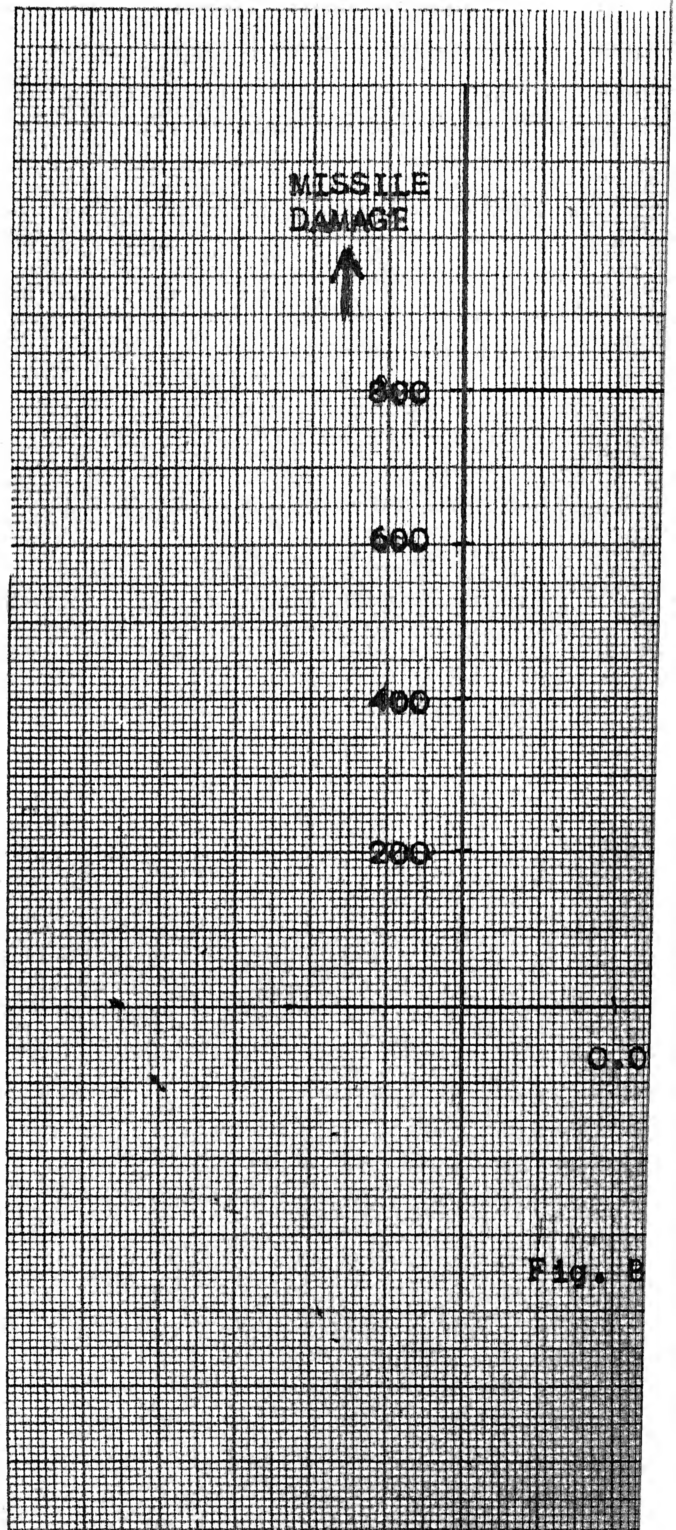
B.1: Gun Damage assessment profile from
simulation run.

TABLE B.2: OUTPUT DATA

FROM MISSILE DAMAGE SIMULATION RUN

MIN/TRA RATIO	MISSILEDAM
.1689005E+00	.8000000E+03
.1754383E+00	.8000000E+03
.1819768E+00	.8000000E+03
.1885146E+00	.8000000E+03
.1950531E+00	.8000000E+03
.2015910E+00	.7872725E+03
.2081288E+00	.7349698E+03
.2146673E+00	.6826620E+03
.2212051E+00	.6303594E+03
.2277436E+00	.5780515E+03
.2342814E+00	.5257490E+03
.2408192E+00	.4734464E+03
.2473577E+00	.4211385E+03
.2538955E+00	.3688359E+03
.2604340E+00	.3165283E+03
.2669718E+00	.2642255E+03
.2735096E+00	.2119231E+03
.2800481E+00	.1596151E+03
.2865859E+00	.1073127E+03
.2931237E+00	.5501031E+02
.2996616E+00	.2707481E+01
.3061994E+00	.0000000E+00
.3127372E+00	.0000000E+00
.3192750E+00	.0000000E+00
.3258129E+00	.0000000E+00
.3323507E+00	.0000000E+00

STOP 0
OK>



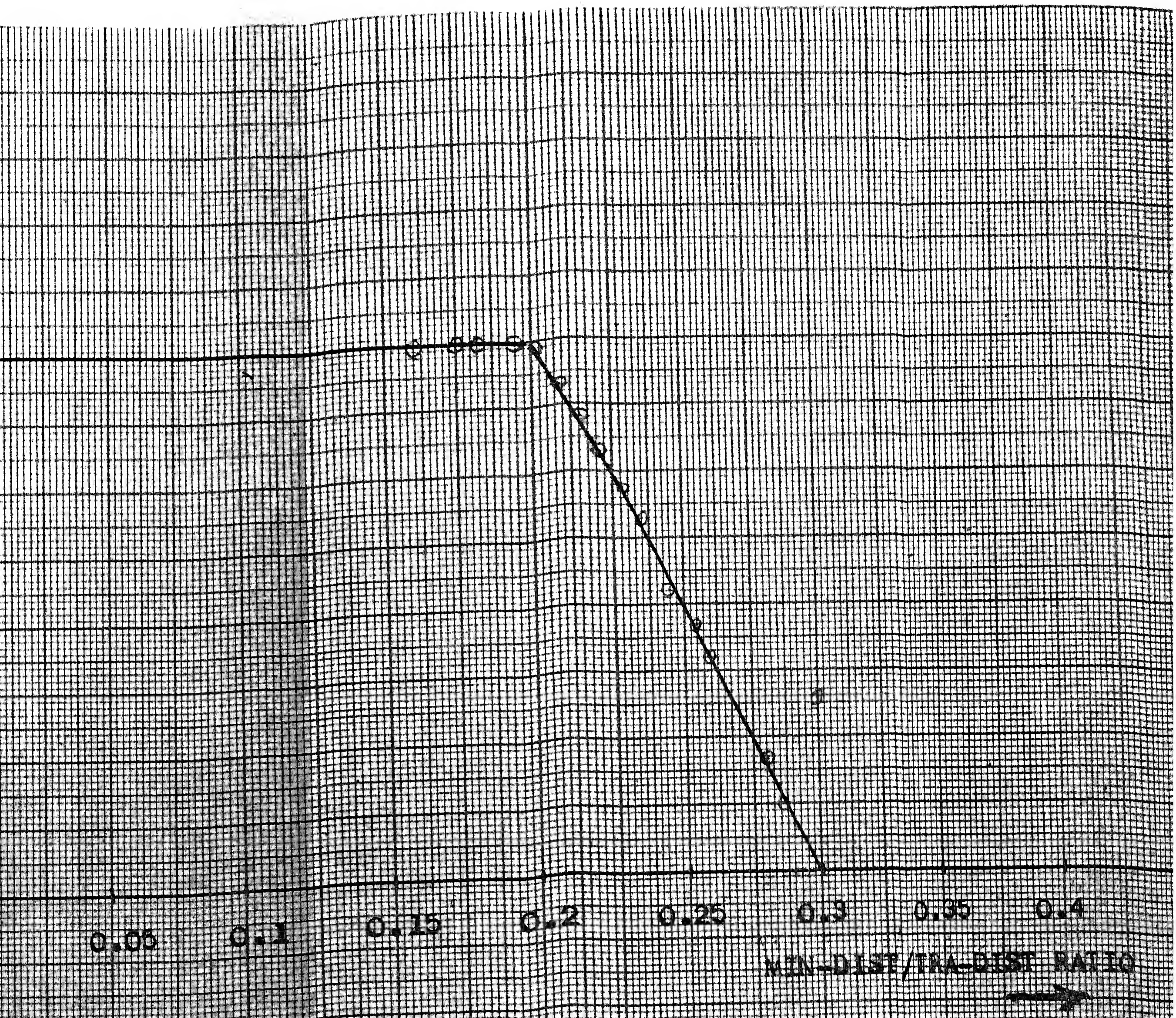


Fig. B.2: Missile Damage Assessment Profile
from Simulation Run.

OK>

TABLE B.3: OUTPUT DATA

ROM BOMB DAMAGE SIMULATION RUN

HIT_DIST	BOMBDAM
.002655	600.00
.027161	600.00
.051666	600.00
.076172	600.00
.100677	600.00
.125153	600.00
.149658	600.00
.174164	600.00
.198669	600.00
.223175	553.65
.247681	504.64
.272156	455.69
.296661	406.68
.321167	357.67
.345673	308.65
.370178	259.64
.394653	210.69
.419159	161.68
.443665	112.67
.468170	63.66
.492676	14.65
.517181	.00
.541656	.00
.566162	.00
.590668	.00

STOP 0
OK>

OK>

OK>

OK>

OK>

OK>

OK>

OK>

OK>

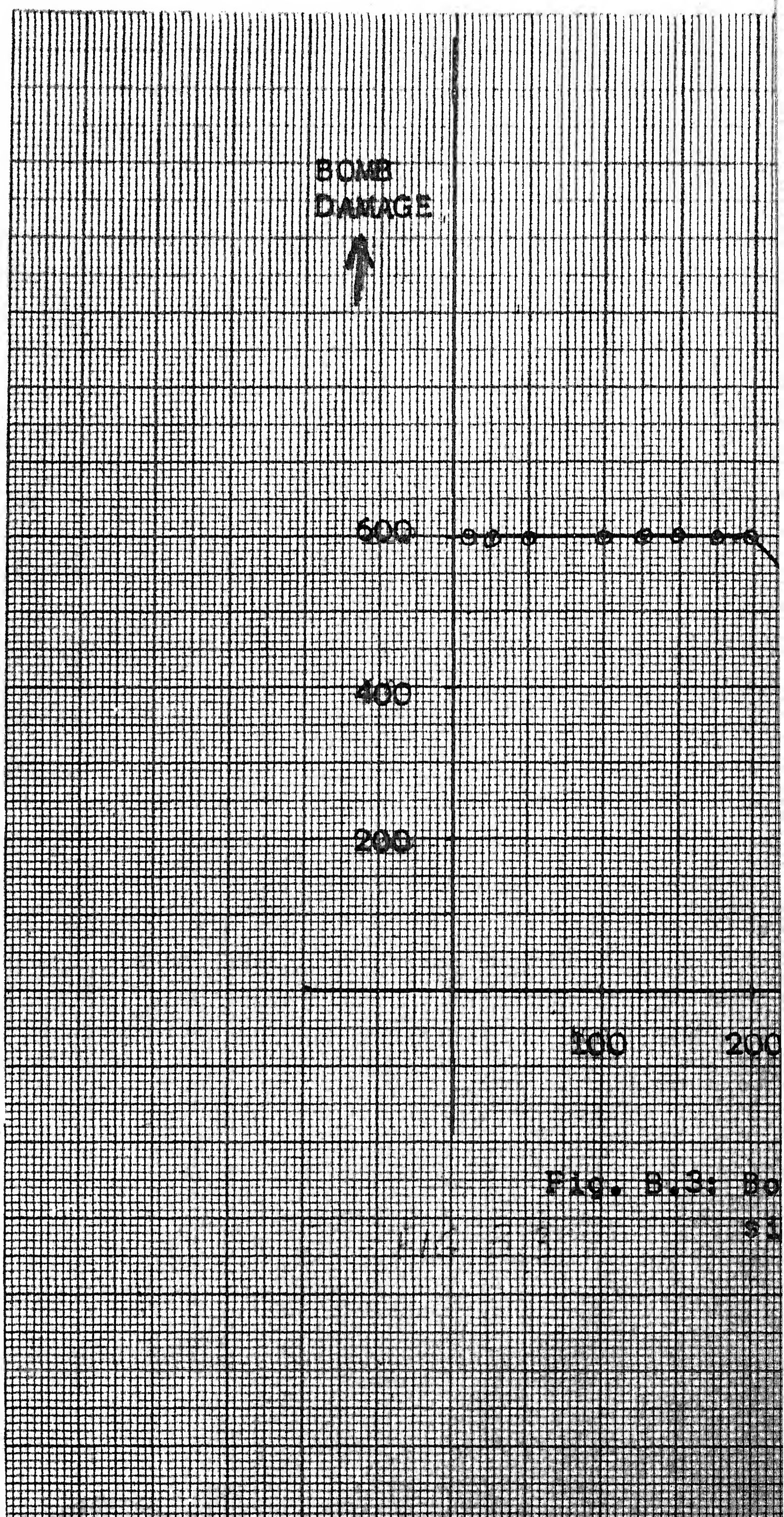
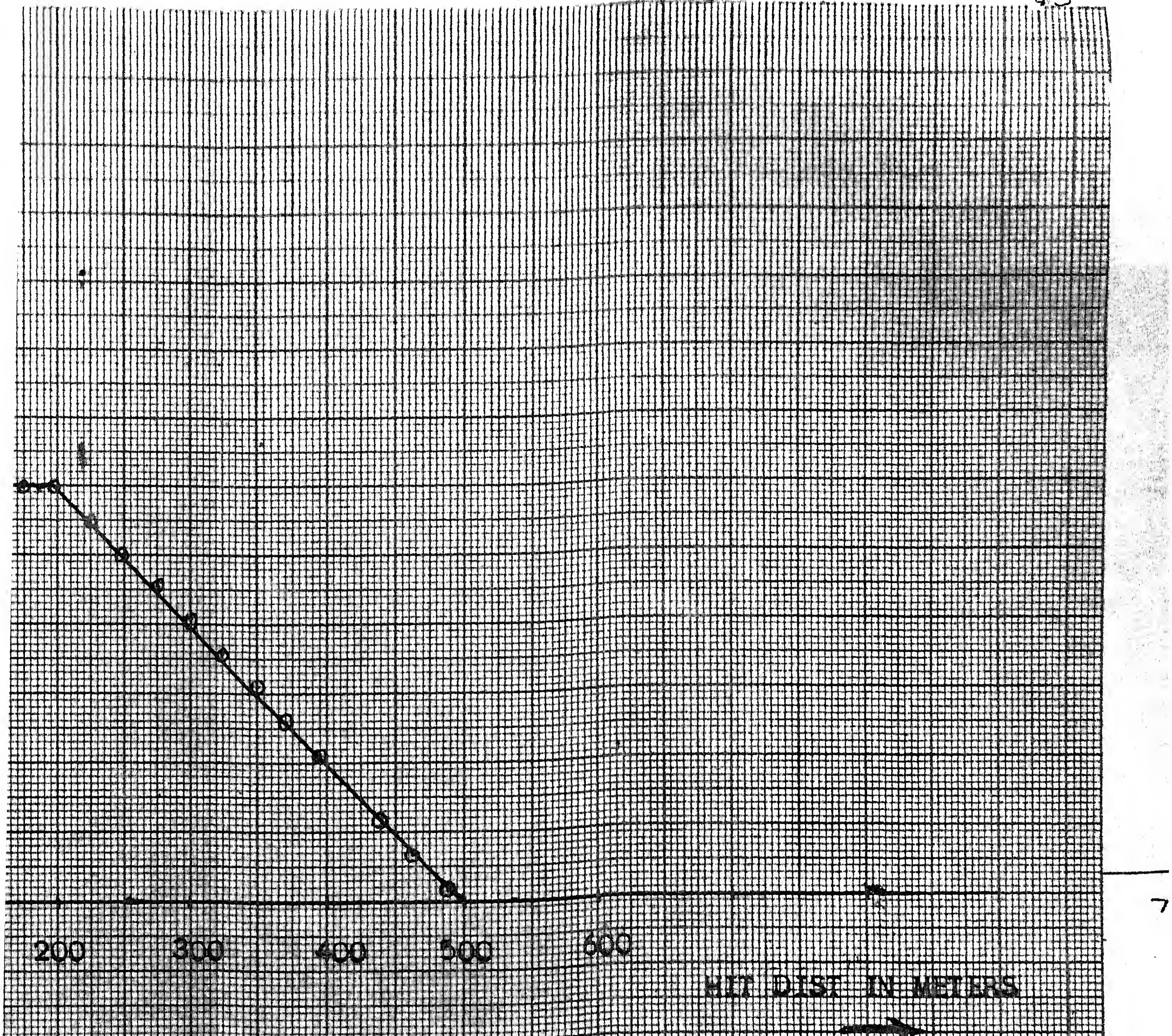


Fig. B.3: Bo



3: Bomb damage assessment profile from
simulation run.

REFERENCES

[1] Banerjee U.K., Operations Analysis and Indian Defence, Concept Publishing, New Delhi, 1980.

[2] Featherstone Donald F., Airwar Games Stanley Paul, London, 1966.

[3] Evans, Wallace and Sutherland, Simulation using Digital Computers, Prentice Hall, Englewood Cliffs, New Jersey, 1967.

[4] Martin Francis, Computer Modelling and Simulation, John Wiley, New York, 1968.

[5] Fishman George S., Concepts and Methods in Discrete Event Digital Simulation, John Wiley, New York, 1973.

[6] Simulation Reference, P.67, Computer Centre Library, I.I.T. Kanpur.

[7] Foley J.D. and Vandam A., Interactive Computer Graphics, Addison Wesley, California, London, 1984.

[8] Colman Andrew, Game Theory and Experimental Games, Pergamon Press, Oxford, 1982.

[9] Owen Guillermo, Game Theory, Academic Press, New York, London, 19

10] Kermode A.C., Mechanics of Flight, Himalayan Books, New Delhi, 19

11] Von-Mises R., Theory of Flight McGraw-Hill, New York, 1945.

12] Kandel A. and Lee S.C., Fuzzy Switching and Automata, Crane Russak, New York, London, 1979.

[13] Kaufmann A., Theory of Fuzzy Subset, Academic Press, New York, London, 1975.

[14] ND PLOT-10/GKS Manual, NORSK DATA, OSLO, 1983.

[15] ND Time-Sharing/Batch Guide, NORSK DATA, OSLO, 1983.

[16] ND SINTRAN-III, Reference Manual, NORSK DATA, OSLO, 1983.

[17] ND FORTRAN-77, Manual, NORSK DATA, OSLO, 1983.

[18] ND PED User's Manual, NORSK DATA, OSLO, 1983.

[19] ND Symbolic Debugger, NORSK DATA, OSLO, 1983.

[20] Graphic Terminal 4105/4107/4109 User Guide, TEKTRONIX, Inc., OREGON, 1985.